A COMPUTER SIMULATION OF AN AIRCRAFT PENETRATION OVER HOSTILE TERRAIN

JAMES L. HARRISON

A COMPUTER SIMULATION OF AN AIRCRAFT

PENETRATION OVER HOSTILE TERRAIN

* * * * *

James L. Harrison

A COMPUTER SIMULATION OF AN AIRCRAFT PENETRATION OVER HOSTILE TERRAIN

by

James L. Harrison

Captain, United States Marine Corps Reserve

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE
IN
OPERATIONS RESEARCH

United States Naval Postgraduate School
Monterey, California

1 9 6 5

A COMPUTER SIMULATION OF AN AIRCRAFT
PENETRATION OVER HOSTILE TERRAIN

by

James L. Harrison

This work is accepted as fulfilling
the thesis requirements for the degree of
MASTER OF SCIENCE
IN
OPERATIONS RESEARCH
from the
United States Naval Postgraduate School

ABSTRACT

A computer simulation of an aircraft making a low level penetration over hostile terrain is described. The model discussed is intended as a tool for the determination of appropriate defensive tactics for aircraft upon receiving fire from conventional ground weapons. In any computer model of a ground to air combat environment it is essential to have a realistic simulation of the terrain. The method used in this simulation is discussed and shown to be extremely accurate.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

[illegible]

1. Introduction

PENAIR is a time step computer simulation of an aircraft flight over hostile terrain. It is an analytic computer war game in which one aircraft and a maximum of two hundred weapons are the principles. Any piece of actual terrain may be simulated using a contour map of the area to obtain the input parameters.

The simulation is intended as a tool for the determination of appropriate defensive tactics for an aircraft upon receiving fire from conventional ground weapons. Using the characteristics of the aircraft and of the weapons the model determines the survival probability of the aircraft on any of several flightpaths. There are no restrictions on the aircraft's speed, altitude or other characteristics and therefore this model is equally suitable for helicopters or jets.

The aircraft flightpath can be predetermined or it can be generated by the model. Upon receiving fire the aircraft can be programmed to execute any of 18 evasive maneuvers. A comparison of the survival probabilities can then be made thus giving an indication of the appropriate defensive maneuver to employ.

The terrain simulation and generated flightpaths appear to be extremely realistic. The terrain is simulated by fitting, in the least squares sense, a polynomial function of two independent variables (x,y) to values of a dependent variable specified at points on a rectangular grid in the plane of the independent variables. This is accomplished by a separate computer program, TERRAIN, which generates the necessary inputs for PENAIR.

2. Design of the Model

PENAIR is a computer simulation of a single aircraft making a low level penetration over terrain containing hostile small arms and conventional anti-aircraft weapons. Any actual or fictitious terrain may be

simulated. No offensive aircraft weapons are considered although 18 different defensive maneuvers such as increasing speed and diving for the deck may be performed. Ten types of ground weapons may be used with a maximum of 20 weapons per type. The program computes the aircraft's time in view of each weapon, the probability of the aircraft surviving each weapon and the aircraft's total survival probability. The simulation runs in discrete time steps using any time interval desired. The program does not utilize Monte Carlo methods; aircraft survival probability for a specified flightpath over a specified terrain is computed in a single run. Flow chart 1.0 of Appendix I, page 30 , indicates the general logic of the PENAIR program.

PENAIR is written in FORTRAN-63 for the CDC 1604 computer. The PENAIR program is included as Appendix II. A separate program, TERRAIN, is utilized to supply inputs to PENAIR for terrain simulation. The terrain input program supplies the necessary data on punched cards for the PENAIR program. The TERRAIN program is also programmed in FORTRAN-63 for the CDC 1604 and is included as Appendix IV.

The terrain used in the PENAIR program may be divided into as many as 12 separate squares for simulation. Each of the 12 individual squares is approximated by a separate polynomial. The 12 squares may be of any common size. Figure 1 illustrates the arrangement of these squares. The coordinate system has been established such that the lower left corner is the origin with increasing x to the right and increasing y upward. It is possible to use actual military grid coordinates since the origin may take any desired x and y values.

A special PENAIR output is a set of punched cards to be used as input to a graph plot routine for an off-line plotter. This routine then draws the actual flightpath over the simulated terrain. Output from this routine is included in Section 6.

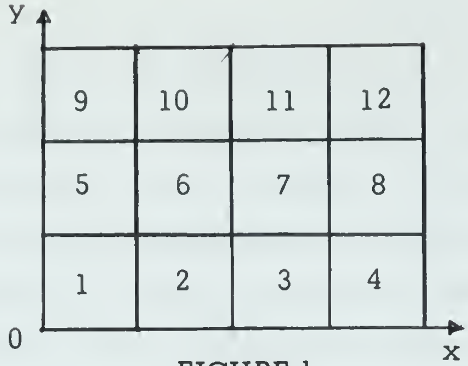


FIGURE 1

THE ARRANGEMENT OF THE INDIVIDUAL
SQUARES OF TERRAIN SIMULATED IN
THE PENAIR PROGRAM

3. Aircraft Movement

The PENAIR model has provisions for three types of simulated aircraft flightpaths. The first of these flightpaths allow the PENAIR program to select one of 18 different evasive maneuvers for the aircraft upon receiving fire. The 18 evasive maneuvers consist of six basic maneuvers with three possible speed changes for each. The six basic maneuvers are: (1) continue flying straight ahead at the same altitude, (2) commence a climb straight ahead at the aircraft's maximum rate of ascent, (3) turn away from the weapon remaining at the same altitude, (4) turn away and commence a maximum rate climb, (5) dive for the deck and commence nap of the earth (contour) flight straight ahead, or (6) turn away from the weapon, dive for the deck and commence nap of the earth flight. For each of the above maneuvers the aircraft may decrease speed, remain at the same speed or increase speed. All of the above maneuvers are generated using the input characteristics of the aircraft being simulated. It should be realized by the PENAIR user however that only one basic maneuver

may be employed in one computer run. The maneuver employed is determined by the numerical value of the input flag IMARK.

The other two simulated aircraft flightpaths are preplanned. In the first of these, the flightpath is completely determined in advance and read into the PENAIR program from data cards. In the second, the x and y coordinates of points along the flightpath are read in to the PENAIR program. From these points the PENAIR program computes a nap of the earth flightpath for the aircraft over the simulated terrain. In both of these cases, the PENAIR program makes no deviation from the pre-planned flightpath and the aircraft makes no use of the evasive maneuvers. A complete description of the inputs needed to regulate aircraft movement is included in Appendix III.

4. Determination of the Aircraft's Survival Probability.

The major output of the PENAIR program is the probability of aircraft survival. This survival probability is computed as a function of weapon type, range and time in view. The necessary inputs are weapon characteristics and the single time-interval kill probabilities for each weapon. Given these kill probabilities, the probability of an aircraft surviving any single weapon is computed as:

$$P\ (\text{Survival}) = (1 - P_1)\ (1 - P_2)\ \ldots(1 - P_n)$$

where P_i is the probability of kill for one time interval at range i for a given weapon. The total probability of aircraft survival is then computed as the product of the individual probabilities of survival for each weapon. (See Flow Chart 5.1, Appendix I, page 48). It is assumed that the probability of a kill by one weapon is independent of all other weapons. The number of time intervals the aircraft receives fire depends on the time in view, minimum and maximum weapon range, and the weapon acquisition, firing, and reloading times.

Time in view is the one parameter to be determined. The method used by this simulation to determine time in view will be explained using Figure 2 and Flow Chart 7.0 of Appendix I, page 49.

The aircraft is considered in view of a weapon if every point on the line of sight WA between the weapon position W, and the aircraft A, is higher than the corresponding point of the terrain. To determine this the x,y coordinates of the point on the line TR, 10 meters from the weapon are used to calculate the elevation of the terrain at that point. This elevation is then compared with the height of the line of sight at those coordinates. If the terrain is lower, another point 50 meters farther along the line TR is chosen and the same comparison is made. If the terrain elevation is below the line of sight elevation at this point this procedure is repeated every 50 meters. This process is continued until it is determined that the aircraft is in view of the weapon or that the terrain elevation exceeds the line of sight. If the latter occurs the aircraft is considered not in view of this weapon at this time. This process is repeated for all weapons and then the aircraft is advanced one time interval. The number of consecutive time intervals the aircraft remains in view of any one weapon is then used to compute the current total amount of time in view per weapon.

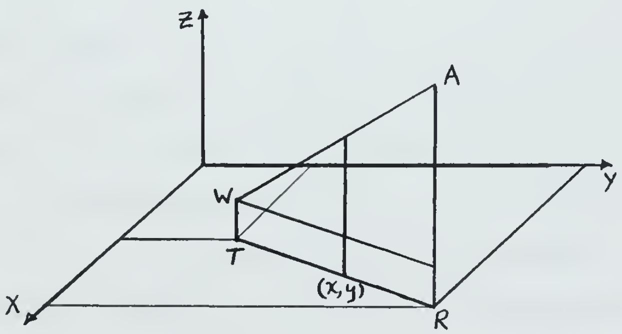


FIGURE 2

DETERMINATION OF THE AIRCRAFT'S TIME IN VIEW OF A WEAPON

Two other methods of determining if the aircraft was in view of the weapon were tested. Elevation comparisons between line of sight and terrain were made at 25 meter increments on the line of sight WA, and at 25 meter increments along TR. No significant differences were noted in the times of view computed by the 3 methods mentioned. However, the method outlined in the previous paragraph greatly reduces computer running time for the PENAIR program.

After the aircraft has been in view of a weapon for the required acquisition time, the weapon minimum and maximum ranges are checked. If the aircraft is within firing range, the weapon commences firing and the probability of survival is determined for this weapon in this time interval. The probabilities of aircraft survival for each weapon are then computed for all following time intervals until the aircraft either goes out of view and/or range or until the weapon has expended its ammunition. In both cases no further probabilities are computed until the weapon is reloaded or until the aircraft comes back in view and within range. It is assumed that after reloading reacquisition requires one-half the acquisition time. If the aircraft goes out of view the weapon must wait its full acquisition time after the aircraft is again in view.

5. Terrain Simulation.

In any computer model of a ground combat environment it is essential to have a realistic simulation of the terrain in order to determine the mask angles of ground weapon positions. To do this the simulation must provide the elevation of any point in the combat area. Present computer methods of terrain simulation either use very specialized equipment or involve prohibitive amounts of computer storage. The common method is to divide the terrain of interest into equal areas and assign an appropriate elevation to each. The assigned elevation might be the average elevation of the area. In this method it is assumed that all points in a given area, for the purpose of simulation, have the same elevation. This is a

crude approximation of terrain features for large areas. The approximation approaches reality as the size of the areas decreases. However, since the number of the areas considered is inversely proportional to the size of the areas, as the approximation of the terrain approaches reality the computer storage required becomes very large.

For the model developed here the terrain features are expressed in polynomial form, $z=f(x,y)$. The general problem is that of fitting points in three dimensions to a surface represented by the polynomial, $z=f(x,y)$. A method of forming this polynomial using the principle of least squares, given the set of data points $\{(x,y,z)\}$, had been programmed in FORTRAN-60 for the CDC 1604 [2]. This program was converted to FORTRAN-63 and a successful attempt was made in fitting a polynomial to a fairly rough surface (mountainous terrain.) The polynomials that best fit mountainous terrain whose elevations ranged from 800 to 1300 feet have approximately 400 terms with the maximum degree of both x and y near 20.

The program designed to fit the polynomial to the terrain required the input points be from an equally spaced grid in the x,y plane. For this thesis, areas of 1000 meters on a side were arbitrarily chosen from maps of actual terrain. In order that the fit be acceptable on the edges of the grid square, an area of at least 1200 x 1200 meters with the desired 1000 meter square centered inside had to be considered. Consecutive x and y coordinates of each individual contour line were recorded and a program was written to produce equally spaced elevations from the consecutive data points. However, because of the great amount of storage required, these two programs could only be incorporated into one by the use of overlays. An overlay is a feature of FORTRAN-63 designed to increase the use of the core storage of the computer by using magnetic tape as auxiliary storage.

The program to compute $z=f(x,y)$ from the consecutive (x,y) points of actual contour lines consists of a main program and three overlays.

The sole purpose of the main program is to call the individual overlays into operation. The first overlay, MAPPING, computes the equally spaced elevation points using the (x,y) points from the contour lines as input. The second overlay, SURFIT, by using the points computed in the first overlay, computes the polynomial $z = f(x,y)$ and records the coefficients on punched cards. The second overlay also computes the elevations of a grid for use in the third overlay. The third overlay, PLOTTER, generates data and stores it on a magnetic tape suitable for a CDC-160 program used to drive a CalComp 165 graph plotter. The contour map drawn can then easily be compared to the original map to estimate the goodness of fit of the polynomial. The drawing of this map is optional and is controlled by an input parameter to the main program. A detailed description of each overlay follows.

Overlay 1 determines the elevation of the terrain at equally spaced points of a grid in the x,y plane. The program was written by T. K. Rodeburg of the Norwegian Defense Institute for Stanford Research Institute, Fort Ord, California. This overlay uses as input the lists of the x and y coordinates at successive points on each contour line read from an enlarged map of the actual terrain. The following steps are taken by this overlay in the process of determining the elevation of the grid points:

(1) The point coordinates are corrected for distortion caused by unequal magnification of the original map.

(2) Equally spaced vertical lines are determined and the coordinates of the intersection of the contour lines and these vertical lines are calculated.

(3) The elevation at these intersections are calculated by linear interpolation.

(4) By further linear interpolation, elevations at equally spaced points on the vertical lines are found, thus obtaining the elevations of points on a grid in the x,y plane.

(5) Using the original data points, the above interpolation procedure is repeated at equally spaced horizontal lines.

(6) The average of the two values thus obtained is used as the elevation at each point of the grid.

Overlay 2 fits, in the least squares sense, a polynomial function of two independent variables, x and y, to values of a dependent variable, z, specified at points on a rectangular grid in the plane of the independent variables. That is, given a set of ground elevations, z at grid coordinates x and y, this overlay finds the polynomial, $\sum_{n=o}^{N} \sum_{m=o}^{M} A_{nm} X^n Y^m$, which best fits the data. The use of orthogonal polynomials leads to a particularly simple system of linear equations rather than the ill-conditioned system which arises from the usual least squares normal equations. It also provides a measure of the improvements resulting from each new term included which further leads, in this algorithm, to an automatic selection of a "best" degree polynomial function as determined by Gauss' criterion [1]. It therefore determines the optimum values of N and M as well as the coefficients.

This algorithm was programmed in FORTRAN-60 for the CDC 1604 by Mrs. Jean Bow of the USNPGS from an ALGOL program [2]. Minor modifications were made to convert it to a FORTRAN-63 overlay. The FORTRAN-60 version is a very general program whereas the overlay has had many of the generalities removed for ease of handling inputs which in turn reduced the computer running time significantly. A FORTRAN listing of this program appears in Appendix IV.

The following steps are taken by Overlay 2 in the process of determining an approximating polynomial:

(1) the variables are normalized by subtracting off their means,

(2) the orthogonal polynomials are evaluated,

(3) the contribution of each orthogonal polynomial to the minimization of the residuals is evaluated,

(4) Gauss' criterion is applied to determine the degree of polynomial which yields the closest fit to the given data.

(5) the orthogonal polynomial coefficients are evaluated,

(6) the dependent variables are evaluated using the approximating polynomial, and

(7) the terrain elevations utilizing a 50 meter grid are evaluated for input into Overlay 3.

Overlay 3 generates lists of x and y coordinates of successive points on each contour line and stores these coordinates on an output tape in a form suitable for a CDC-160 program used to drive a CalComp 165 plotter. This tape is used as an input tape for the plotter which then reconstructs the terrain by connecting the points with straight line segments.

The FORTRAN-63 program used in Overlay 3 was originally developed for use with the IBM 7090 system and an x,y plotter, to give a complete traditionally styled contour map. This program was converted to FORTRAN-63 by Mr. R. R. Hillery of the USNPGS, for use on the CDC 1604. Subroutines used to compute the points to be plotted were provided by Dr. M. O. Dayhoff of the National Biomedical Research Foundation, Silver Spring, Maryland [3]. Subroutines used internally to write the output tape were programmed by Mr. J. R. Ward of the USNPGS for use with a CalComp 165 graph-plotting routine, DRAW.

Figure 3 is an actual contour map from which approximately 1200 input points have been read. A polynomial of 400 terms was fitted and the elevations of a grid (spacing of 50 meters) were computed to plot Figure 4. The coefficients of this polynomial ranged from 904.18 for the constant term to -4.593×10^{-94} for the $x^{19}y^{19}$ term.

An analysis of a table of differences showed the maximum difference of 625 actual and computed elevations to be 31 feet in an elevation range of 900 to 1200 feet.

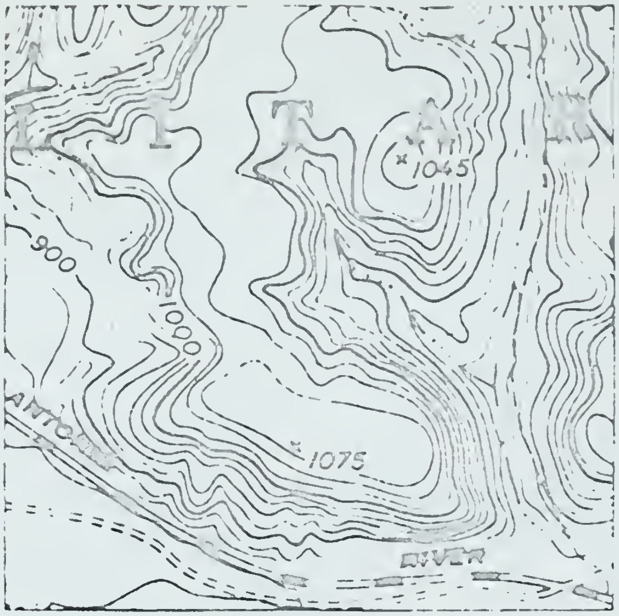


FIGURE 3

ACTUAL CONTOUR MAP OF GRID SQUARE 7174
U. S. ARMY HUNTER-LIGGETT MILITARY
RESERVATION

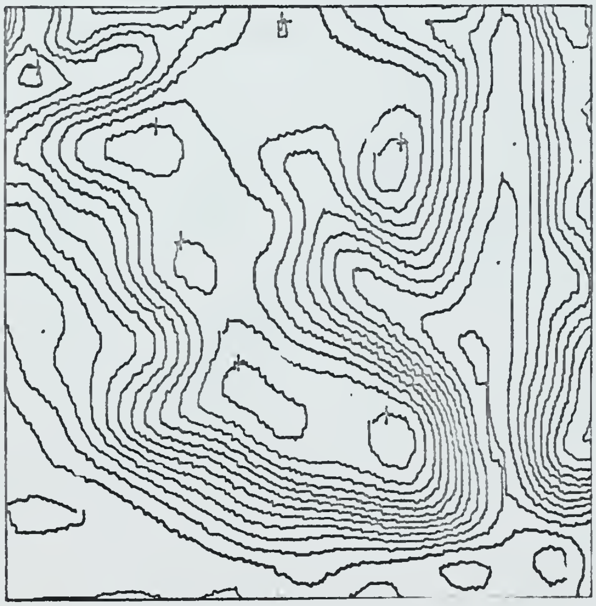


FIGURE 4

COMPUTER SIMULATION OF GRID SQUARE 7174
U. S. ARMY HUNTER-LIGGETT MILITARY
RESERVATION

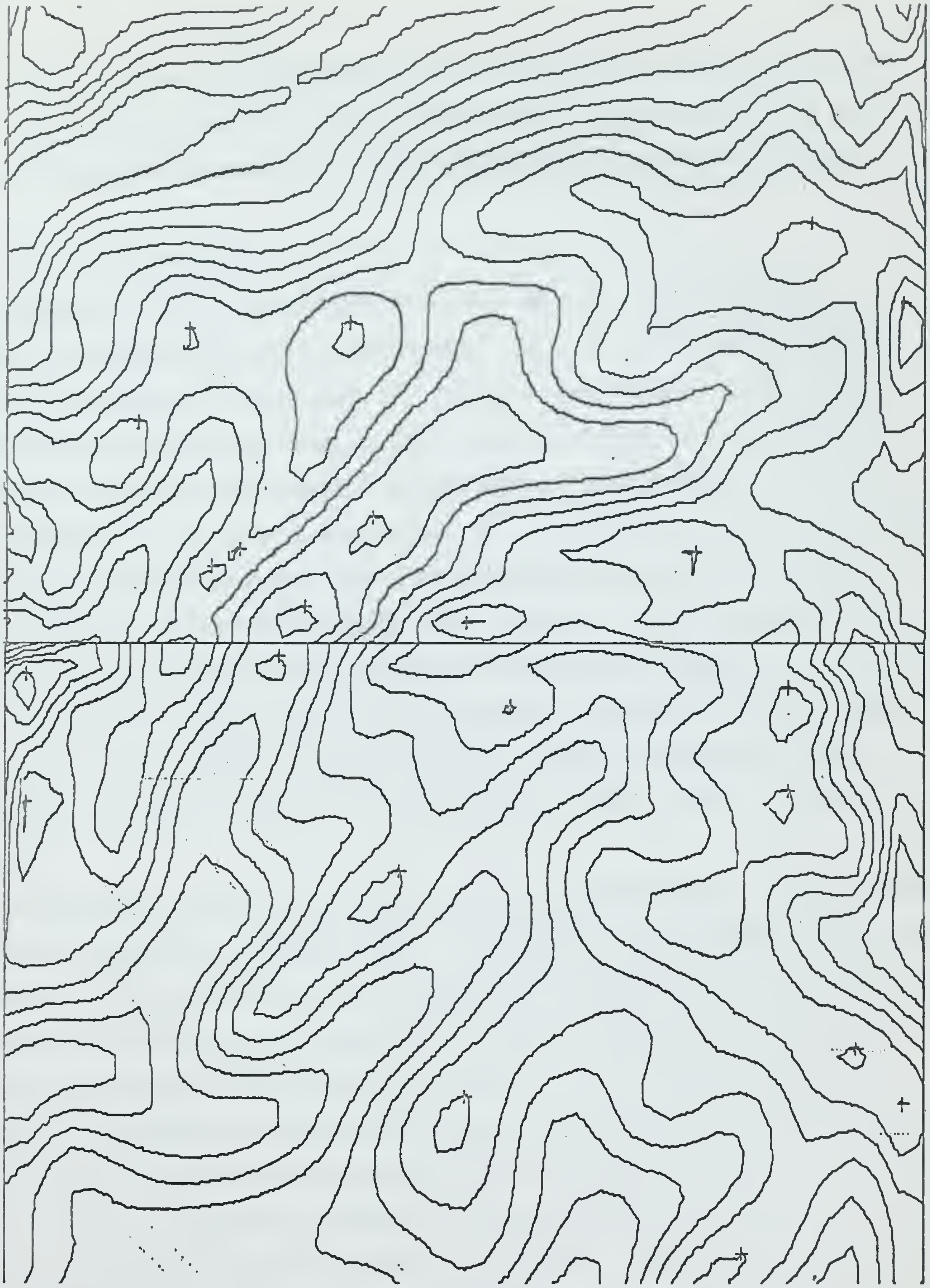


FIGURE 5

REPRESENTATIVE FIT AT THE JUNCTION
OF TWO GRID SQUARES

Figure 5 shows a representative fit at the junction of two of the grid squares approximated by separate polynomials. The accuracy with which the grid squares fit together is determined by the accuracy of the input data.

6. Analysis of the Model

PENAIR has been checked out with the use of a simple scenario. A 3000 x 4000 meter mountainous section of the U. S. Army Military Reservation was selected for simulation. Three 10-men enemy squads armed with M-14 rifles were situated as illustrated in Figure 6. A UH-1B helicopter was assigned the mission of delivering supplies from an airfield at position a to an outpost at position b. Two test runs were made with altitudes of the helicopter of 1000 feet and 500 feet above the terrain. In both tests all 18 evasive maneuvers were checked. Some representative flightpaths generated by the program are illustrated in Figures 6, 7, 8, and 9. The survival probabilities of the aircraft on the various attempts were computed and are shown in Tables 1 and 2. All of the computed probabilities are realistic and are, in a comparative sense, what one would expect.

A sensitivity analysis was then conducted on the rate of climb and descent and changes in aircraft speeds. A helicopter was simulated flying at 100 knots with its maximum rate of climb and descent taking the values of 500,1000 and 2000 feet per minute. The effects of these three rates on nap of earth flight are illustrated in Figure 10. It should be noted that Figure 10 illustrates how the program generates the flightpath to overshoot the crest of a hill before descending just as a pilot tends to do. Figure 10 also shows, as should be expected, the greater the rate of climb and descent, the closer the aircraft remains to the earth. Figure 11 illustrates an aircraft flown with a maximum rate of climb and descent of 1500 feet per minute and speeds of 100, 200 and 300 knots. The resulting flightpaths also appear realistic.

The flightpath computed by PENAIR illustrated in Figure 12 is of a helicopter making a level turn at 70, 90 and 110 knots. Here it is noted, for the computed flightpath, the radius of turn increases as the airspeed increases.

The last analysis made was the comparison of computed results with actual results. Figure 13 is the simulated flightpath and terrain of an actual flightpath flown by a UH-1B over an actual terrain. The 44 weapon positions illustrated were actually manned by U. S. Army personnel. The observed time the helicopter was in view of each of these positions is listed in Table 3. The times in view for the simulated terrain and flightpath were computed and are also listed in Table 3. It is believed that with better terrain data an even closer comparison could be obtained.

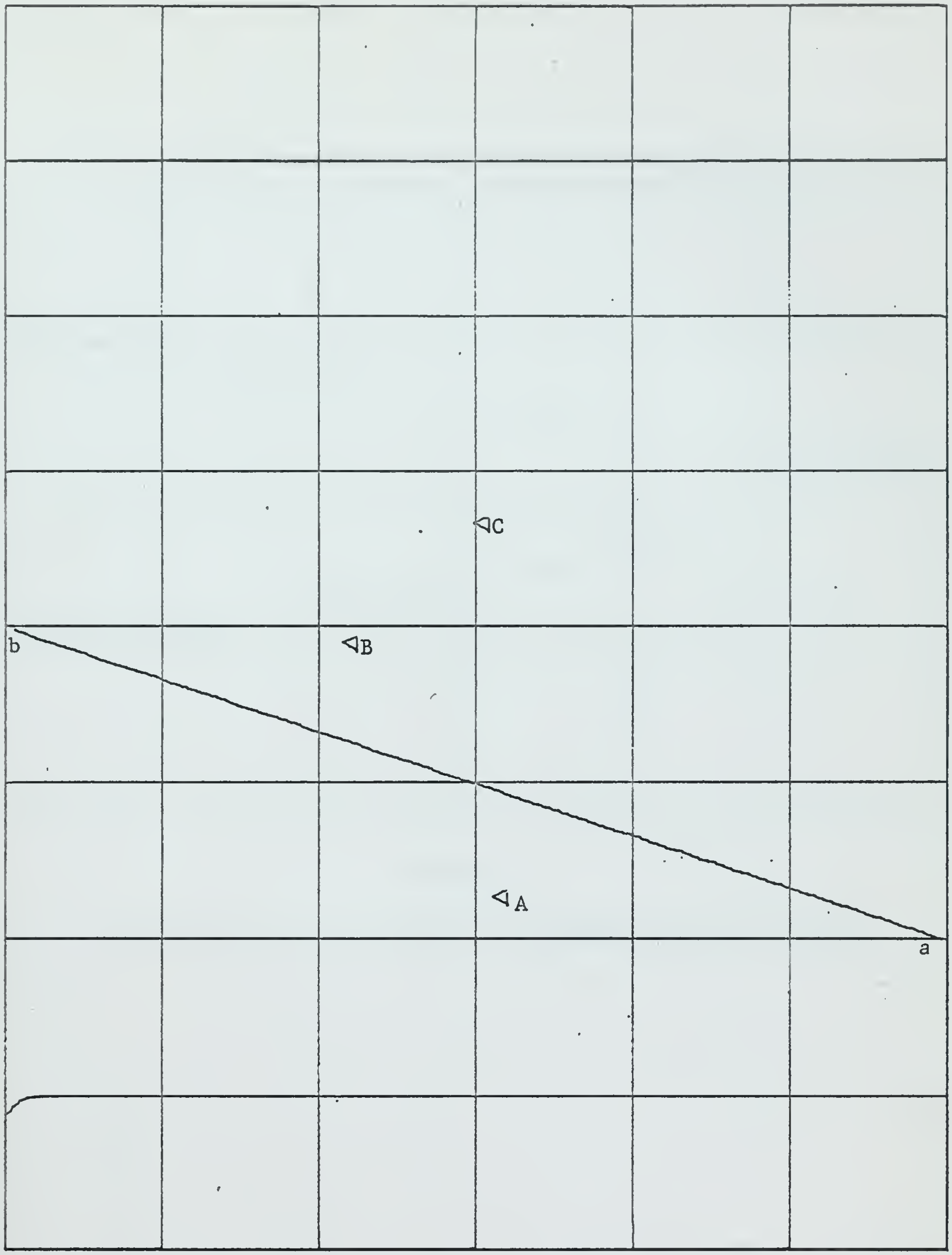


FIGURE 6

A STRAIGHT AND LEVEL FLIGHTPATH

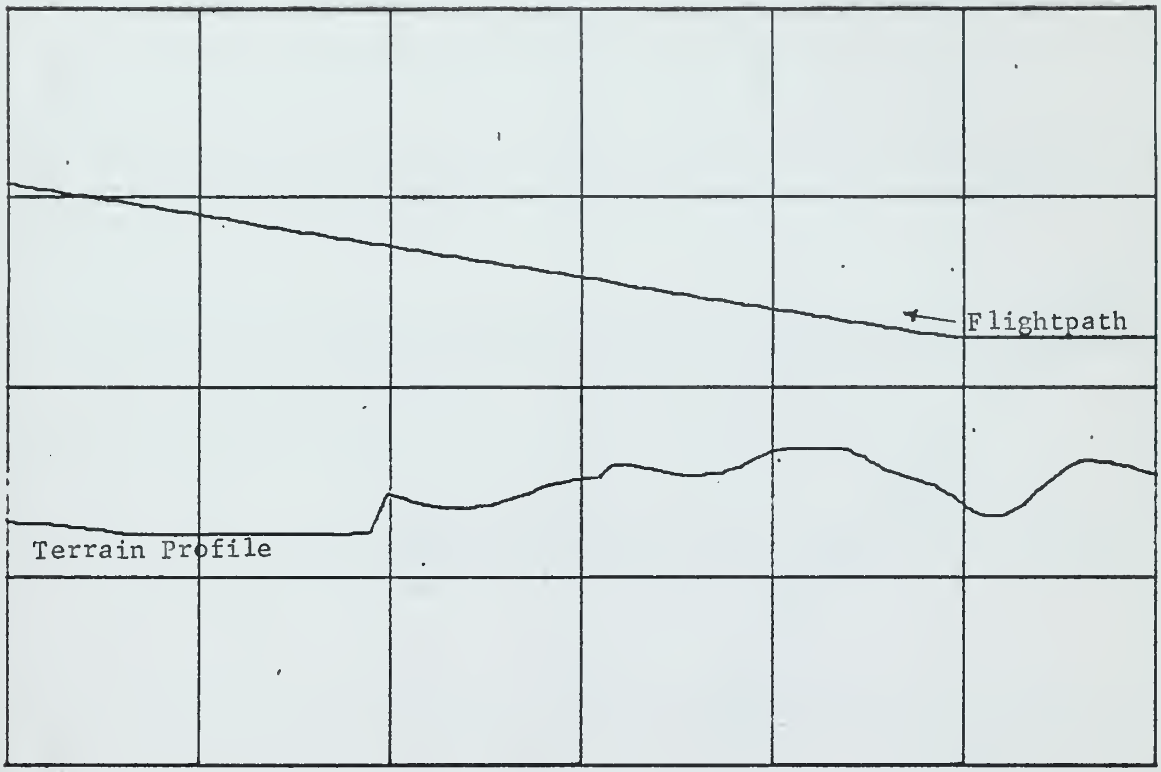


FIGURE 7

A FLIGHTPATH OF AN AIRCRAFT CLIMBING STRAIGHT AHEAD

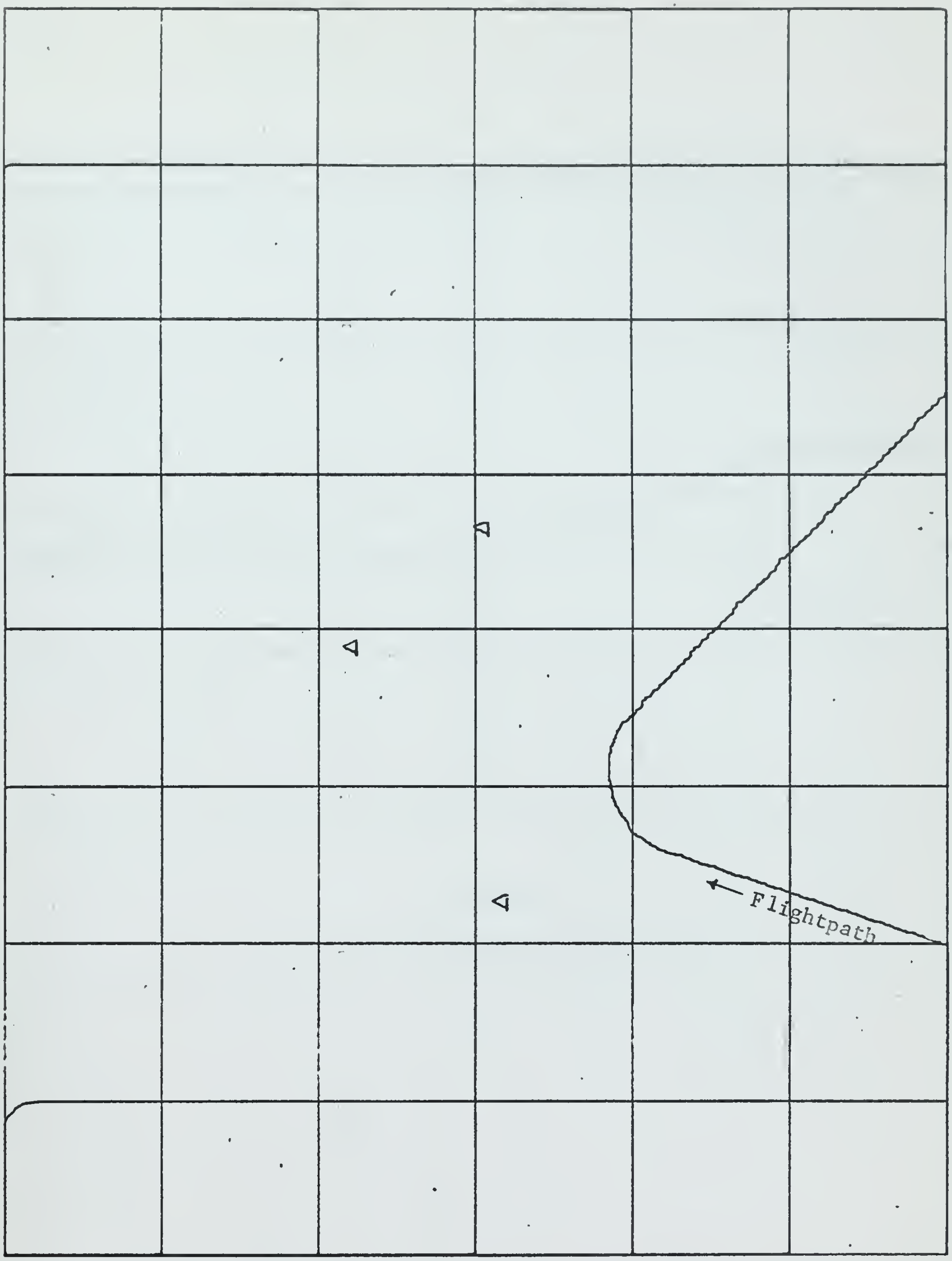


FIGURE 8

A FLIGHTPATH OF AN AIRCRAFT TURNING AWAY FROM A FIRING WEAPON

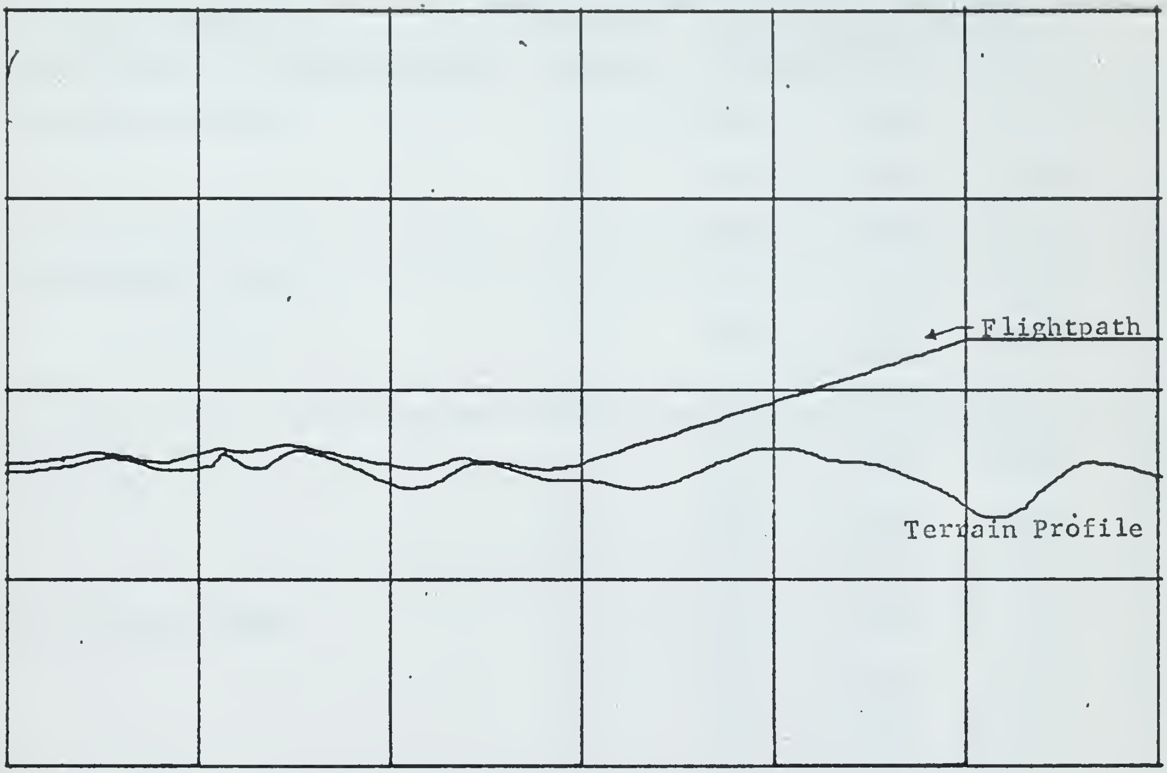


FIGURE 9

A NAP OF THE EARTH FLIGHTPATH

Summary of Survival Probability of A UH-1B Helicopter Fired on by Three 10-Men Squads of M-14 Rifles

Initial Speed 90 Kts — Initial Altitude 650 Meters

Evasive Maneuver (Commenced after receiving fire)	Evasive Speed	Survival Probabilities Wpn A	Wpn B	Wpn C	Total
Fly straight and level	70	.8576	.8078	1.0000	.6928
	90	.8723	.8649	1.0000	.7545
	110	.8723	.8723	1.0000	.7610
Climb straight ahead	70	.8723	.8723	1.0000	.7610
	90	.8723	.8723	1.0000	.7610
	110	.8723	.8723	1.0000	.7610
Turn away, level	70	.9026	1.0000	1.0000	.9026
	90	.9026	1.0000	1.0000	.9026
	110	.9026	1.0000	1.0000	.9026
Turn away and climb	70	.9026	1.0000	1.0000	.9026
	90	.9026	1.0000	1.0000	.9026
	110	.9026	1.0000	1.0000	.9026
Dive and fly NOE	70	.8359	.7252	1.0000	.6062
	90	.8723	.8031	1.0000	.7005
	110	.8723	.8242	1.0000	.7190
Turn away and fly NOE	70	.8798	1.0000	1.0000	.8798
	90	.8874	1.0000	1.0000	.8874
	110	.8874	1.0000	1.0000	.8874

Table 1

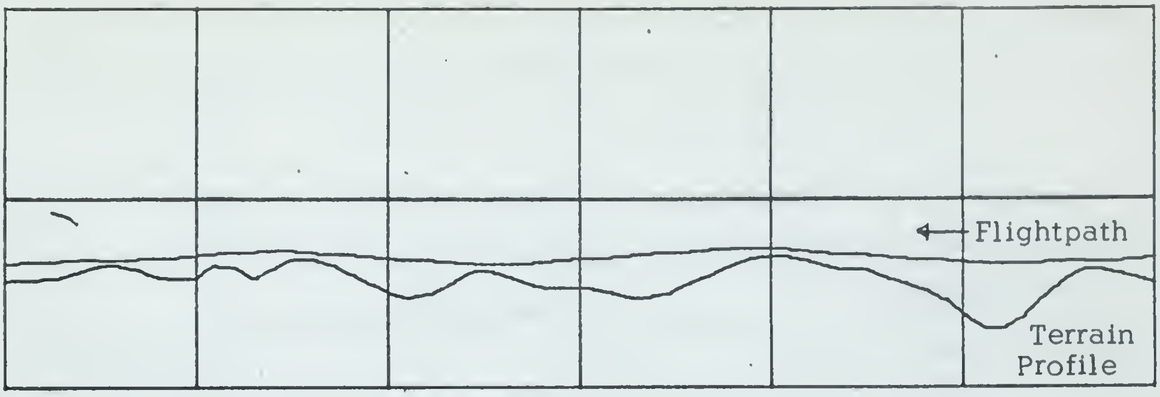
[illegible]

Summary of Survival Probabilities of UH-1B Helicopter Fired on by Three
10-Men Squads Armed with M-14 Rifles

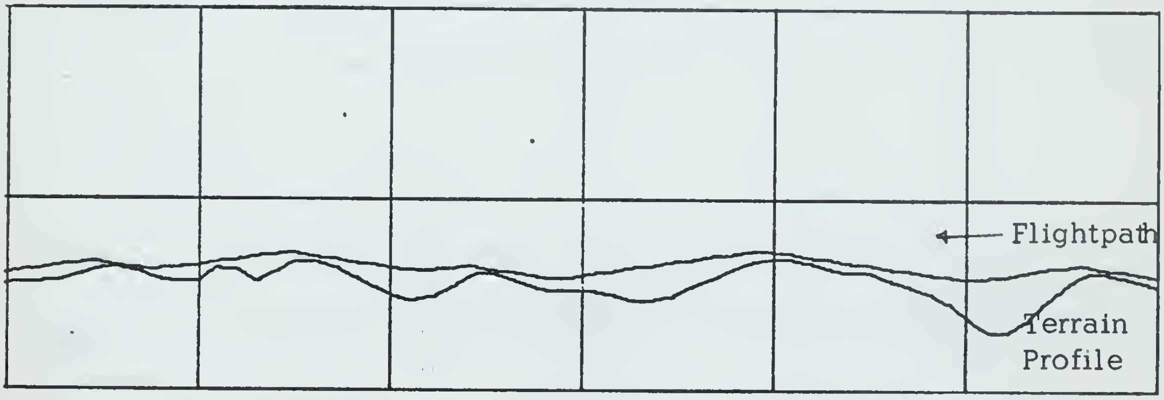
Initial Speed 90 Kts Initial Altitude 450 Meters

Evasive Maneuver (Commenced after receiving fire)	Evasive Speed	Survival Wpn A	Wpn B	Probabilities Wpn C	Total
Fly straight and level	70	Values not determined at these speeds			
	90				
	110	.8309	.7788	1.0000	.6471
Climb straight ahead	70	Values not determined at these speeds			
	90				
	110	.8377	.8242	1.0000	.6905
Turn away, level	70	.8874	1.0000	1.0000	.8874
	90	.8874	1.0000	1.0000	.8874
	110	.8874	1.0000	1.0000	.8874
Turn away and climb	70	.8798	1.0000	1.0000	.8798
	90	.8798	1.0000	1.0000	.8798
	110	.8798	1.0000	1.0000	.8798
Dive and fly NOE	70	.7558	.9501	1.0000	.7181
	90	.8027	.9664	1.0000	.7758
	110	.3306	1.0000	1.0000	.8306
Turn away and fly NOE	70	.8798	1.0000	1.0000	.8798
	90	.8798	1.0000	1.0000	.8798
	110	.8798	1.0000	1.0000	.8798

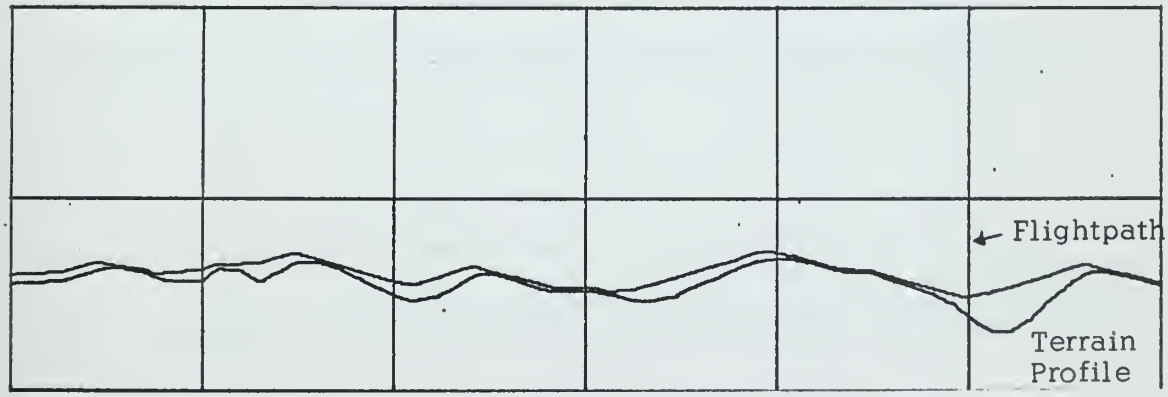
Table 2



RATE OF CLIMB = RATE OF DESCENT = 500 FEET PER MINUTE



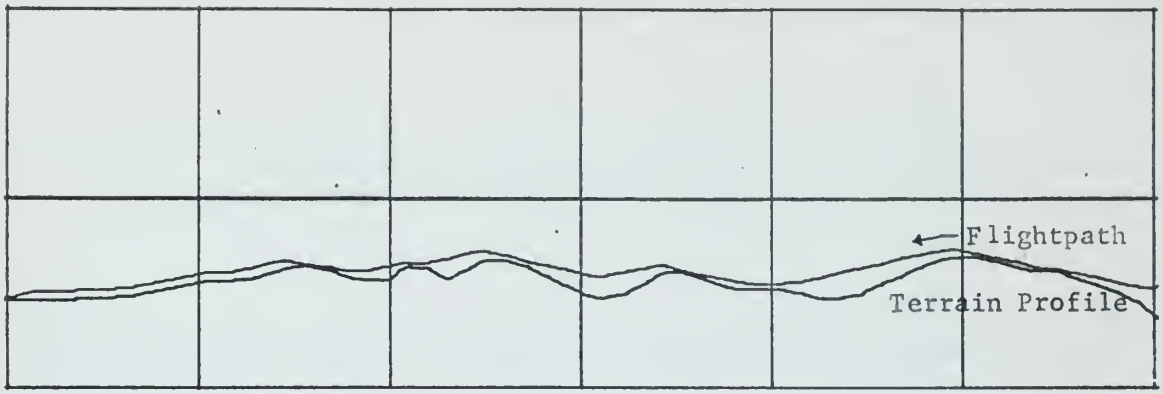
RATE OF CLIMB = RATE OF DESCENT = 1000 FEET PER MINUTE



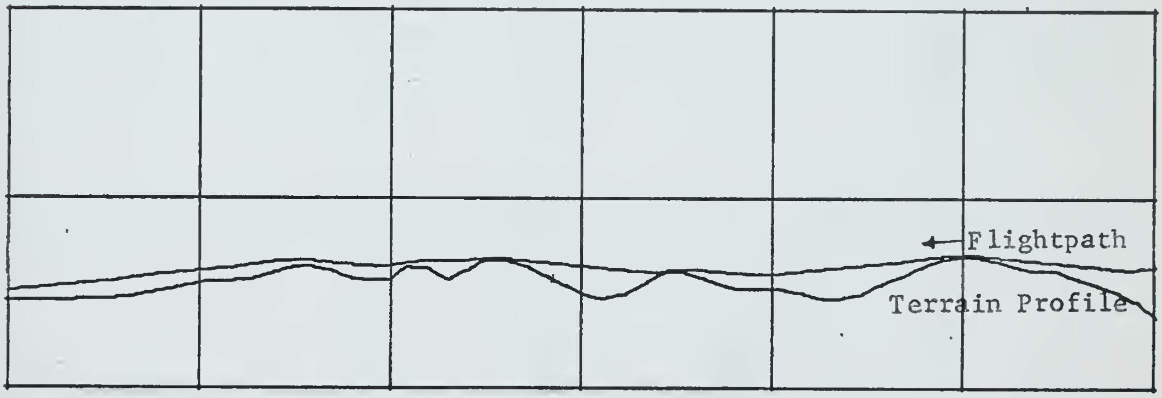
RATE OF CLIMB = RATE OF DESCENT = 2000 FEET PER MINUTE

FIGURE 10

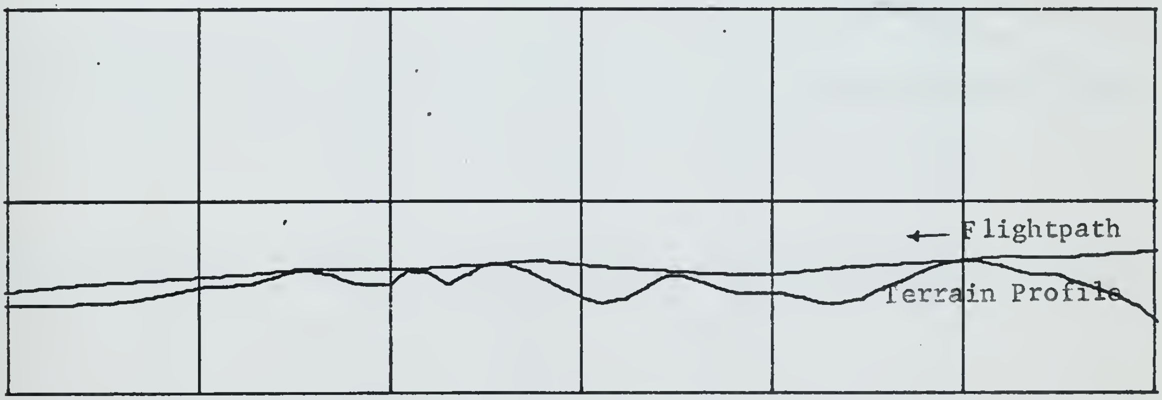
COMPARISON OF RATES OF CLIMBS AND DESCENTS IN A NAP OF THE EARTH FLIGHT AT A CONSTANT SPEED OF 100 KTS.



.100 Kts



200 KTS.



300 KTS.

FIGURE 11.

COMPARISON OF AIRCRAFT SPEEDS ON A NAP OF THE EARTH FLIGHT

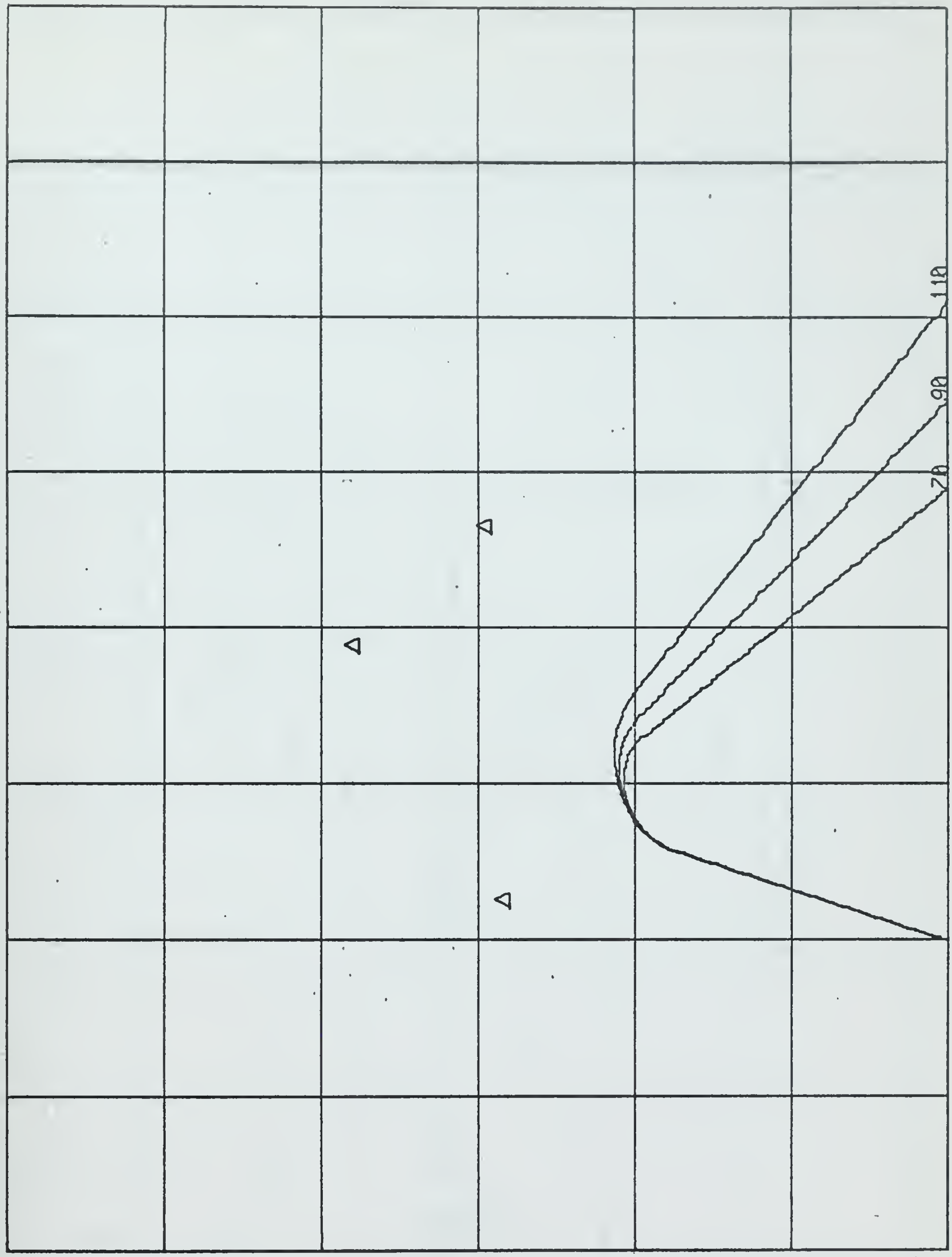


FIGURE 12

COMPARISON OF THE AIRCRAFT'S RADIUS OF TURN
AT 3 AIRSPEEDS, 70, 90 and 110 KTS.

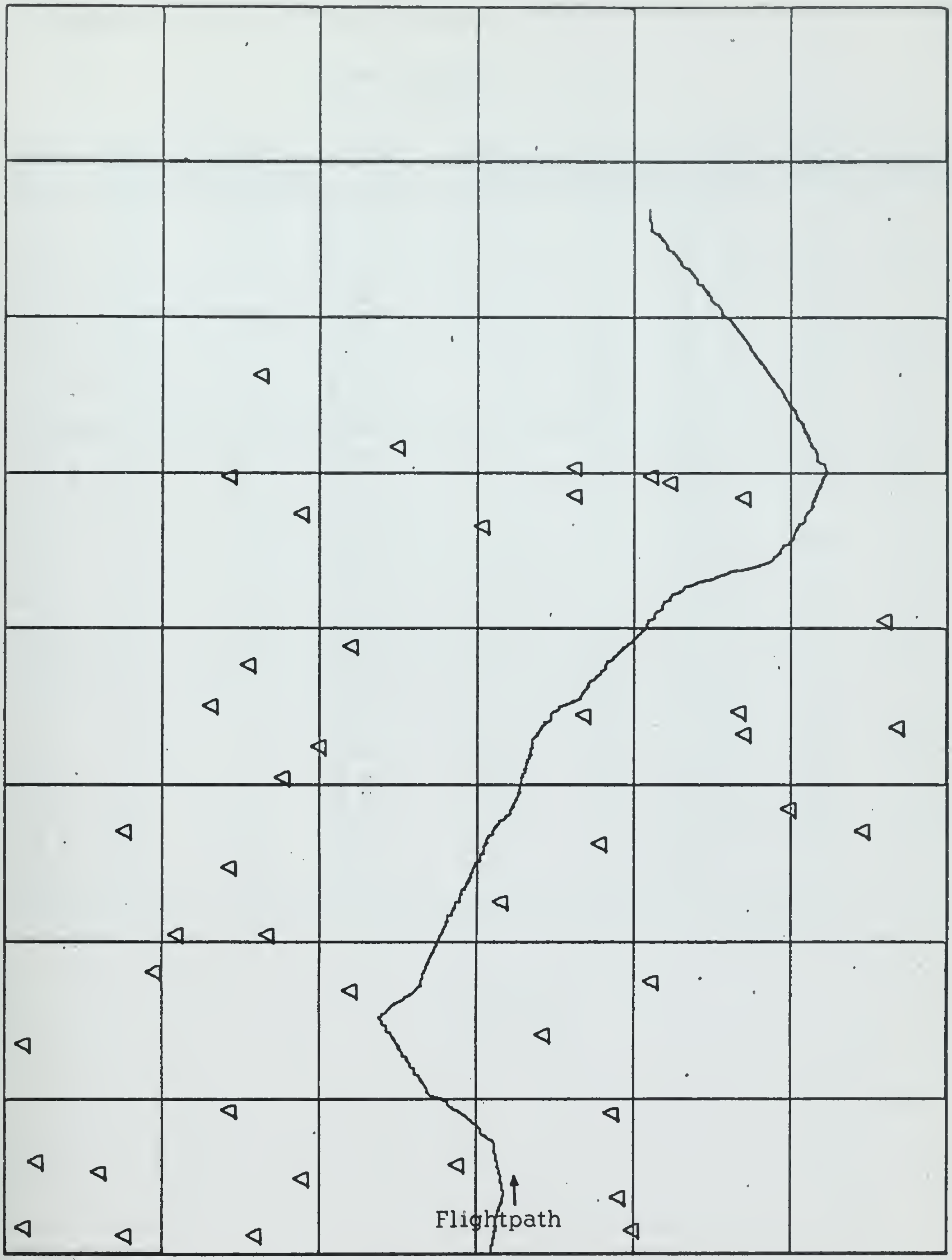


FIGURE 13

COMPUTER SIMULATION OF AN ACTUAL FLIGHTPATH
FLOWN BY A UH-1B HELICOPTER

Comparison of Actual and Computed Time Intervals in View of 44 Weapon Sites

Time Interval = .6 sec.

Wpn. No	Time in View Actual	Time in View Computed	Wpn No	Time In View Actual	Time In View Computed	Wpn No	Time In View Actual	Time In View Computed
1	0	0	16	50	56	31	0	0
2	0	9	17	0	0	32	22	39
3	30	30	18	43	47	33	0	0
4	0	0	19	63	108	34	89	88
5	0	0	20	4	30	35	93	91
6	0	59	21	17	74	36	69	76
7	0	0	22	0	72	37	66	58
8	0	40	23	0	0	38	49	64
9	0	0	24	0	0	39	16	36
10	0	17	25	59	63	40	0	16
11	9	2	26	0	34	41	8	8
12	0	0	27	63	64	42	56	56
13	26	39	28	43	45	43	19	15
14	40	63	29	0	9	44	103	102
15	55	73	30	0	0			

Table 3

7. Conclusions and recommendations.

The simulation of the terrain obtained by this program has proven to be extremely realistic. (See Figure 4, Section 5, page 11). The junctions of squares approximated by separate polynomials are as good as the data used for the approximation. (See Figure 5, Section 5, page 12). This program was written in its present form because of the availability of input data to the author in the format described in Appendix V. If the actual elevations could be obtained using an equally spaced grid for the x and y coordinates, the first overlay could be eliminated. It is felt that this is feasible and would make the preparation of the input cards much simpler.

Among the uses of the simulation of the terrain by this method include ground to ground combat and ground to air combat environments. Mask angles, terrain profiles and actual distances between points in a mountainous terrain are all easily determined. The uses of the PENAIR program include the determination of optimal flightpaths across hostile terrain, the survival probability and optimal maneuvers to perform if the aircraft is fired upon.

The results in Section 6 all tend to illustrate the realism of the aircraft penetration model. Whether the actual kill probabilities used are exact is not extremely important since an appropriate defensive maneuver for an unarmed aircraft to initiate upon receiving ground fire may be obtained by observing the relative survival probabilities of several maneuvers.

It is believed that with additional programming this model could be extended to include formations of two or more aircraft, thus greatly increasing its usefulness. In addition, since the terrain is simulated by mathematical functions and since the line of sight between aircraft and weapon may be described by a linear relationship, it appears possible to solve the two relations for a point of intersection. Upon accomplishing this programming, it is believed the computer running time would be reduced significantly.

BIBLIOGRAPHY

1. Graybill, F. A. An introduction to linear statistical models, volume 1. McGraw-Hill Book Company, Inc., 1961.

2. Clark, R. E., Kubik, R. N., and Phillips, L.P. Orthoganal polynomial least squares surface fit. Communications of the ACM, v. 6, April, 1963: 162-163.

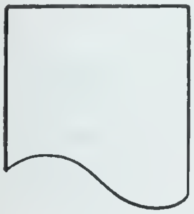
3. Dayhoff, M.O. A contour-map program for x-ray crystallography. Communications of the ACM, v. 6, October, 1963: 620-622.

APPENDIX I

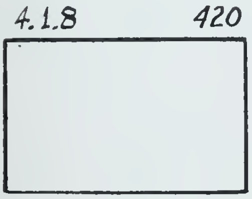
Flow Charts of the PENAIR Program

In the flow charts of this appendix, the following conventions are used:

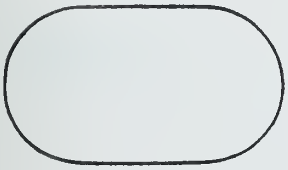
This symbol is used for a READ statement.



This symbol is used for a PRINT statement.



A rectangle indicates one or more FORTRAN statements. A number such as 4.1.8 indicates the statement in this rectangle is further explained by Flow Chart 4.1.8 in this appendix. A number such as 420 is the statement number as it appears in the FORTRAN listing in Appendix II.



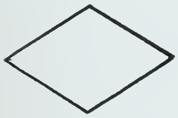
This symbol is used for an IF statement.

This symbol indicates the logic of this Flow Chart is continued at the beginning of Flow Chart 5.1 of this appendix.

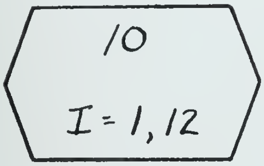
This symbol indicates a connector between two parts of the same Flow Chart of this appendix.

This symbol indicates the logic of this Flow Chart is continued where the same symbol appears in Flow Chart 4.1 of this appendix.

This is the symbol for a CONTINUE statement whose statement number is 10.



This is the symbol for a computed GO TO statement.

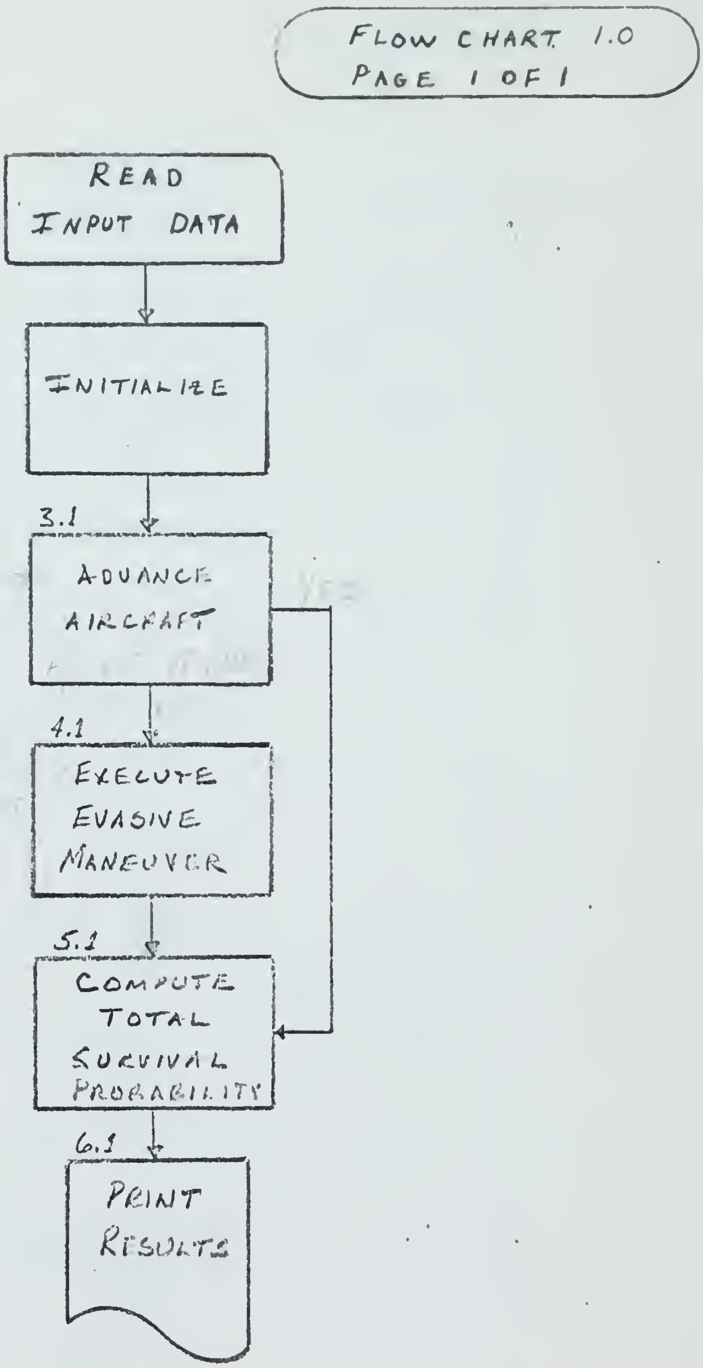


This symbol is used for the statement

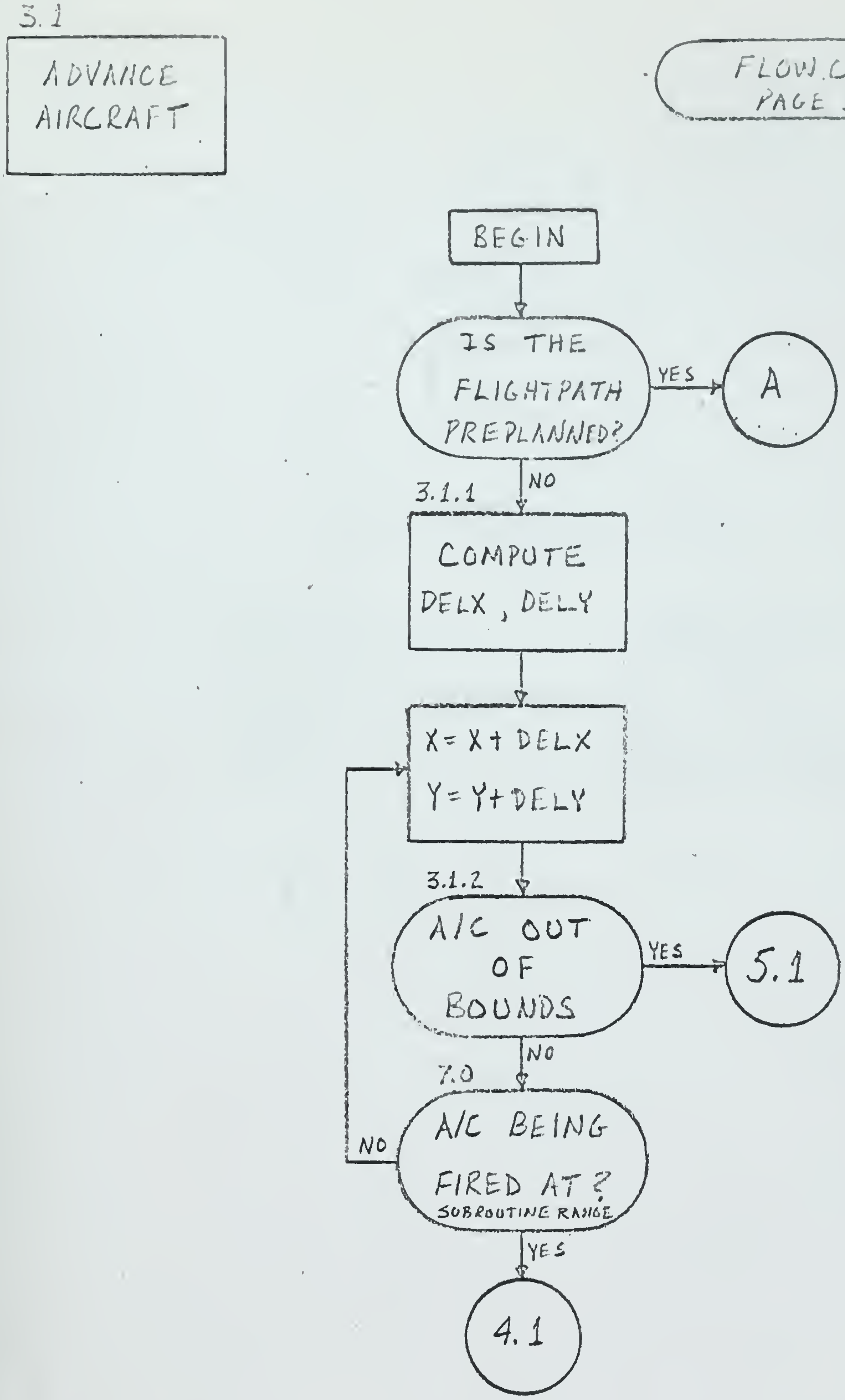
DO 10 I = 1, 12

This symbol indicates the beginning or entry point for the Flow Chart.

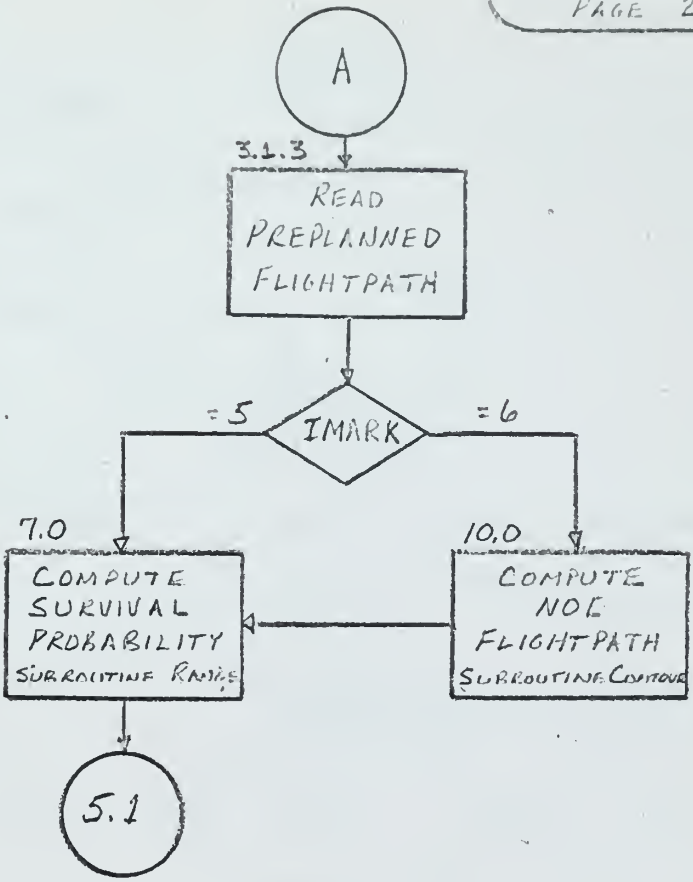
This symbol indicates the end or exit point for the Flow Chart.

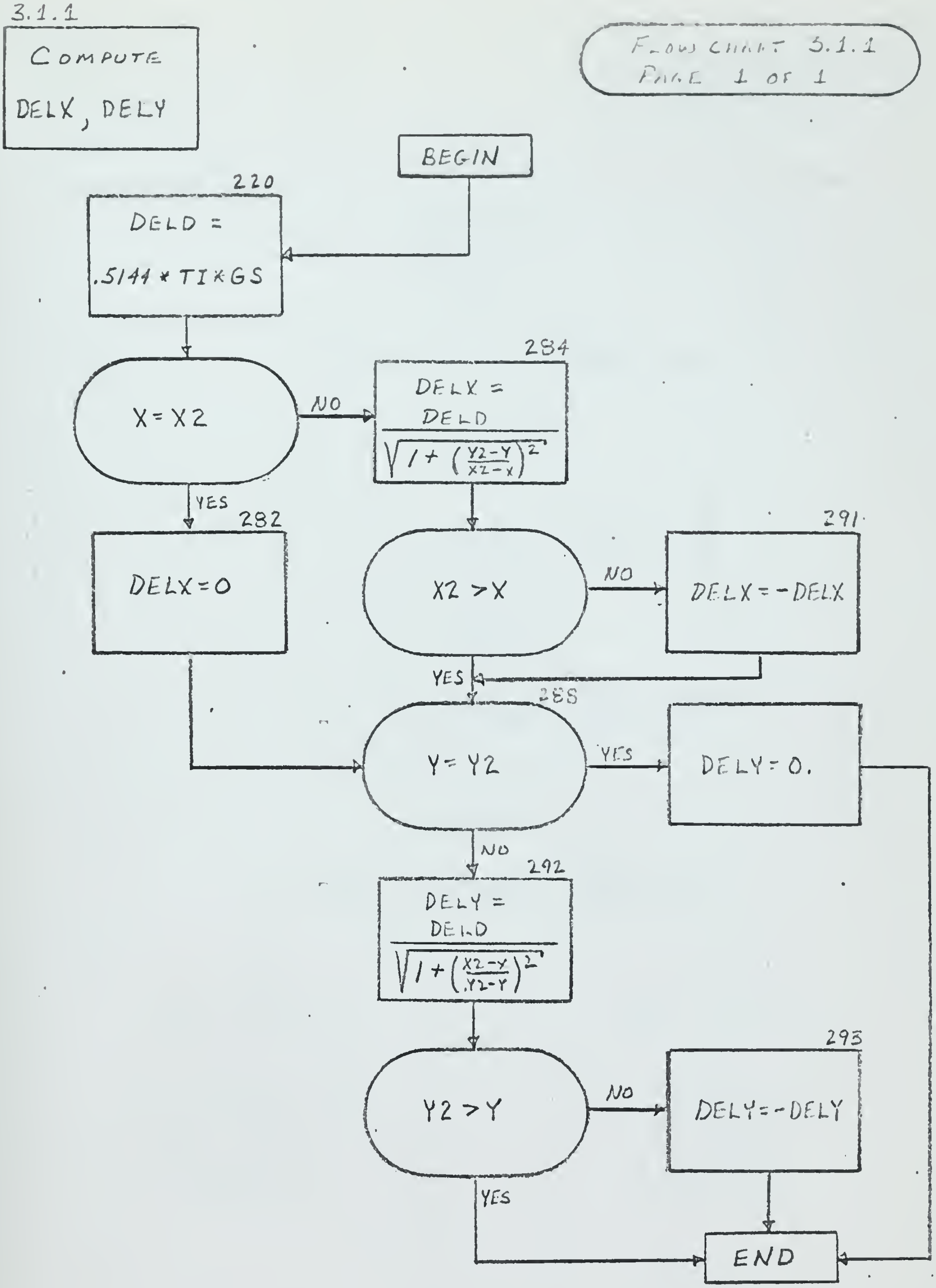


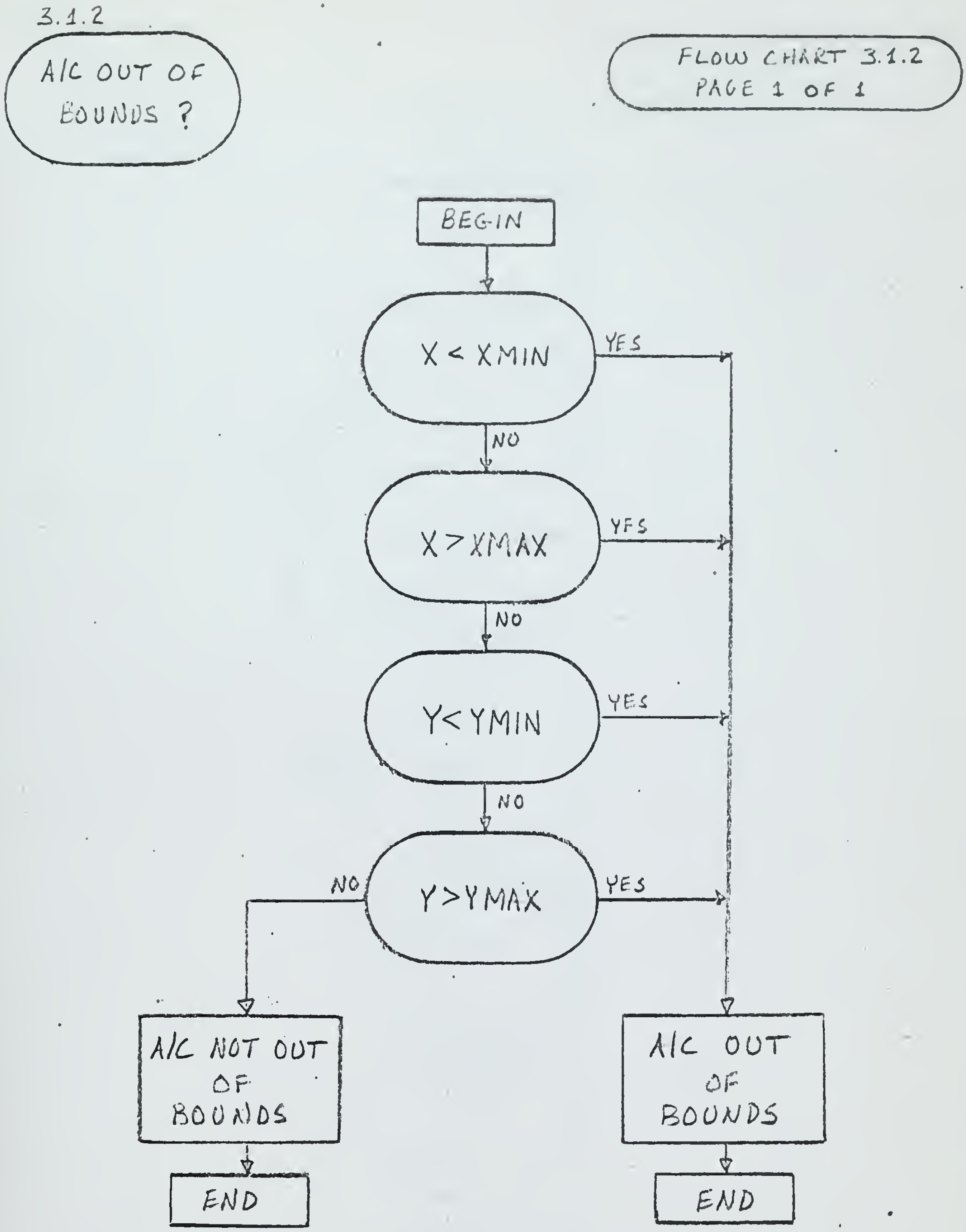
GENERAL FLOW CHART ILLUSTRATING LOGICAL PATHS OF PENAIR.


3.1
ADVANCE AIRCRAFT
BEGIN
IS THE FLIGHTPATH PREPLANNED?
YES
A
NO
3.1.1
COMPUTE DELX, DELY
X = X + DELX
Y = Y + DELY
3.1.2
A/C OUT OF BOUNDS
YES
5.1
NO
7.0
A/C BEING FIRED AT?
SUBROUTINE RANGE
NO
YES
4.1

FLOW CHART 3.1
PAGE 2 OF 2

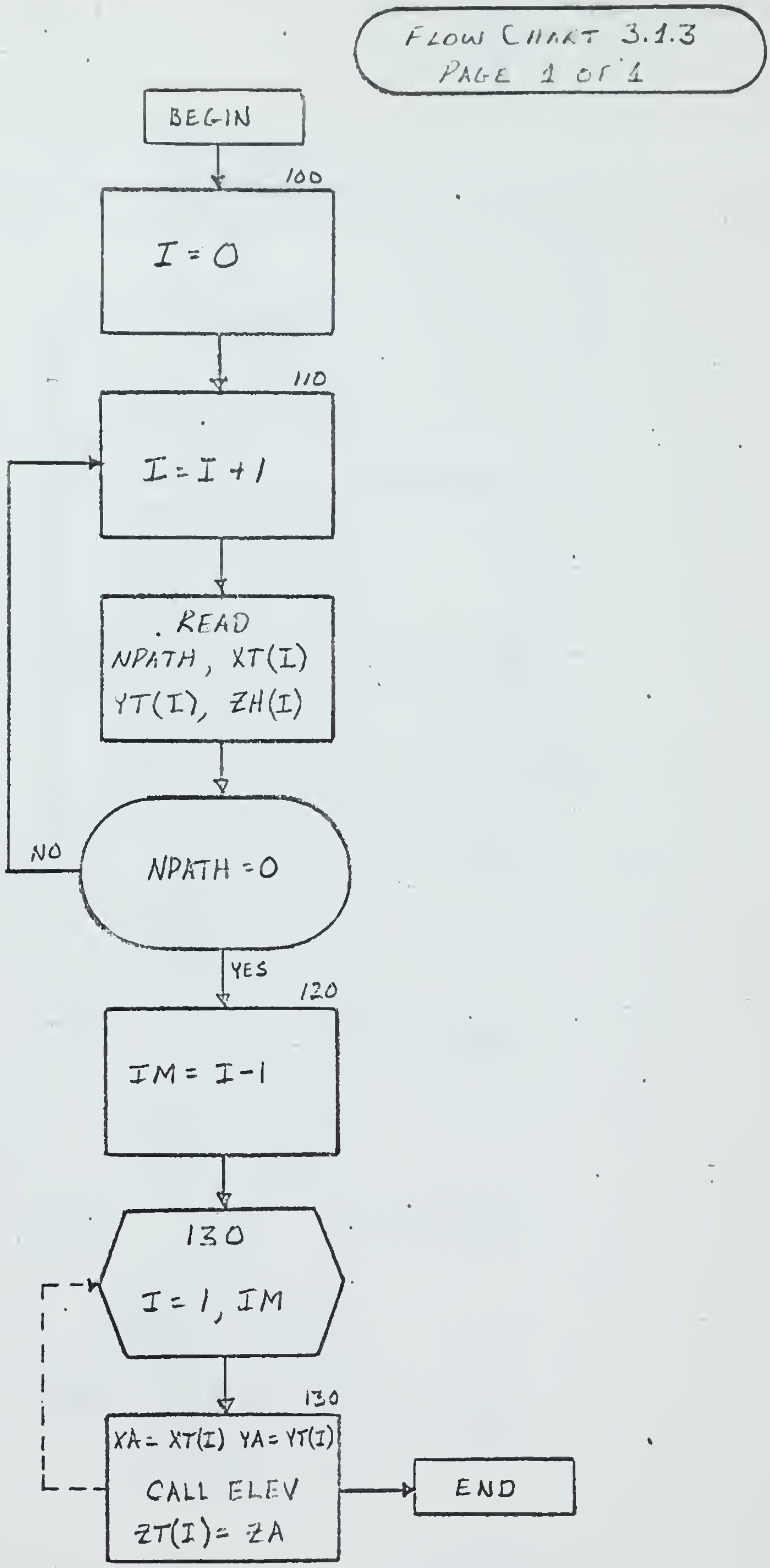



3.1.1
COMPUTE
DELX, DELY
FLOW CHART 3.1.1
PAGE 1 OF 1
BEGIN
220
DELD =
.5144 * TI*GS
X = X2
NO
YES
284
DELX =
$\frac{DELD}{\sqrt{1+\left(\frac{Y2-Y}{X2-X}\right)^2}}$
282
DELX=0
X2 > X
NO
YES
291
DELX=-DELX
285
Y = Y2
YES
DELY=0.
NO
292
DELY =
$\frac{DELD}{\sqrt{1+\left(\frac{X2-X}{Y2-Y}\right)^2}}$
Y2 > Y
NO
YES
293
DELY=-DELY
END


3.1.2
A/C OUT OF BOUNDS ?
FLOW CHART 3.1.2
PAGE 1 OF 1
BEGIN
X < XMIN
YES
NO
X > XMAX
YES
NO
Y<YMIN
YES
NO
Y>YMAX
NO
YES
A/C NOT OUT OF BOUNDS
A/C OUT OF BOUNDS
END
END

3.1.3

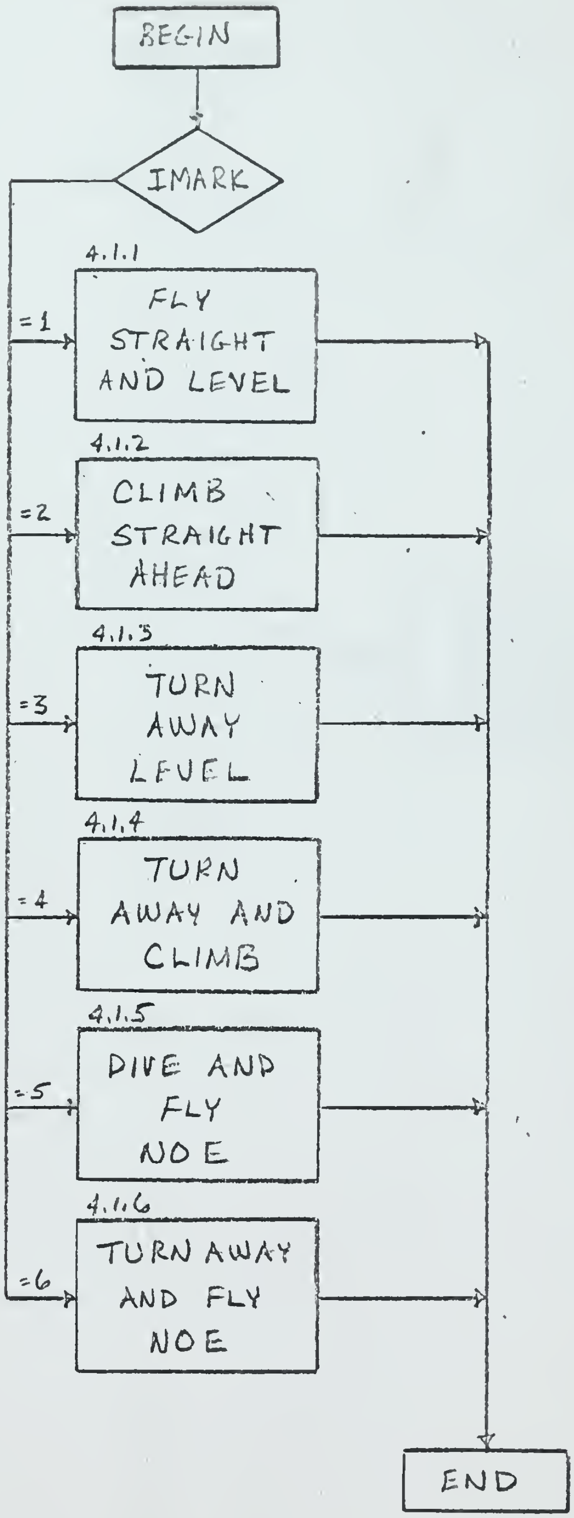
READ PREPLANNED FLIGHTPATH



4.1

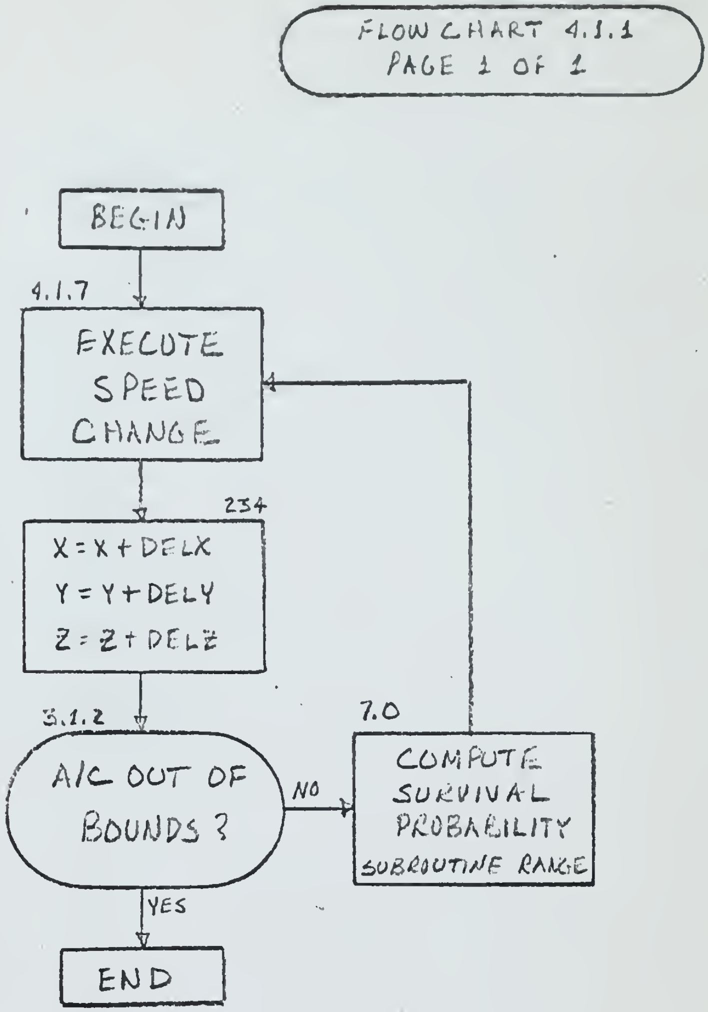
EXECUTE EVASIVE MANEUVER

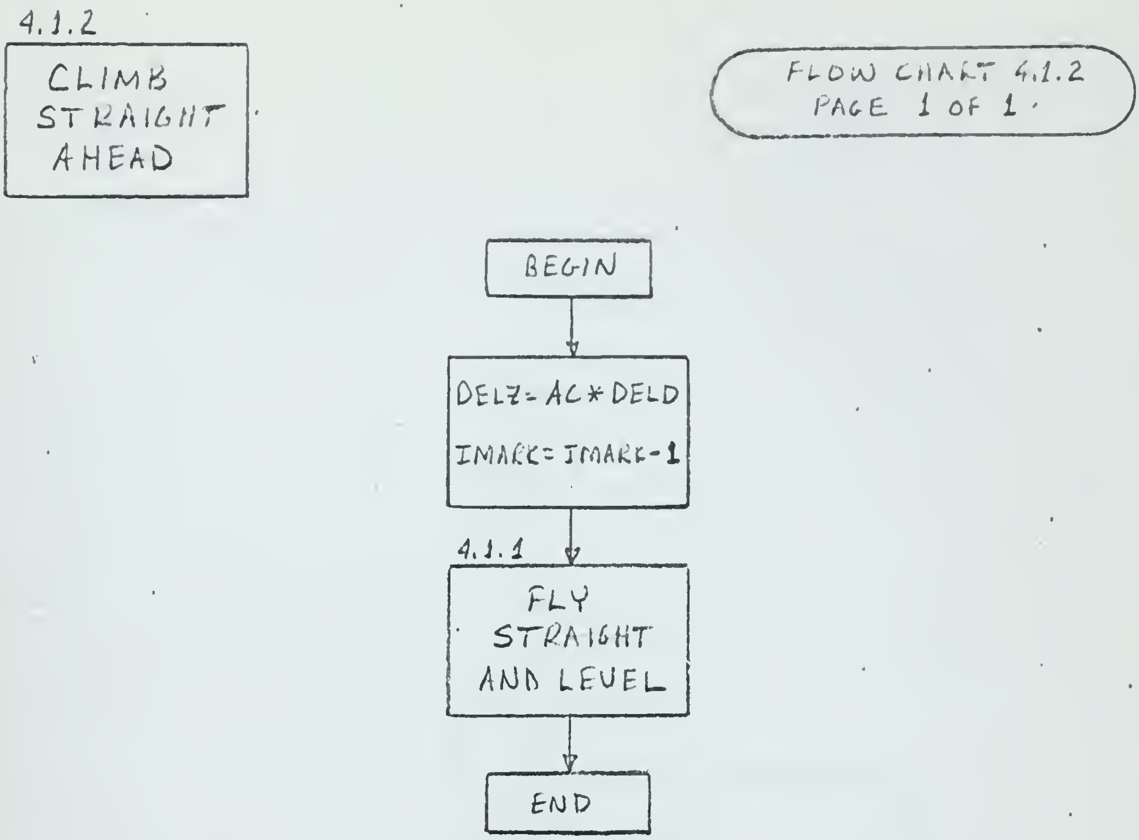
FLOW CHART 4.1
PAGE 1 OF 1



4.1.1

FLY STRAIGHT AND LEVEL

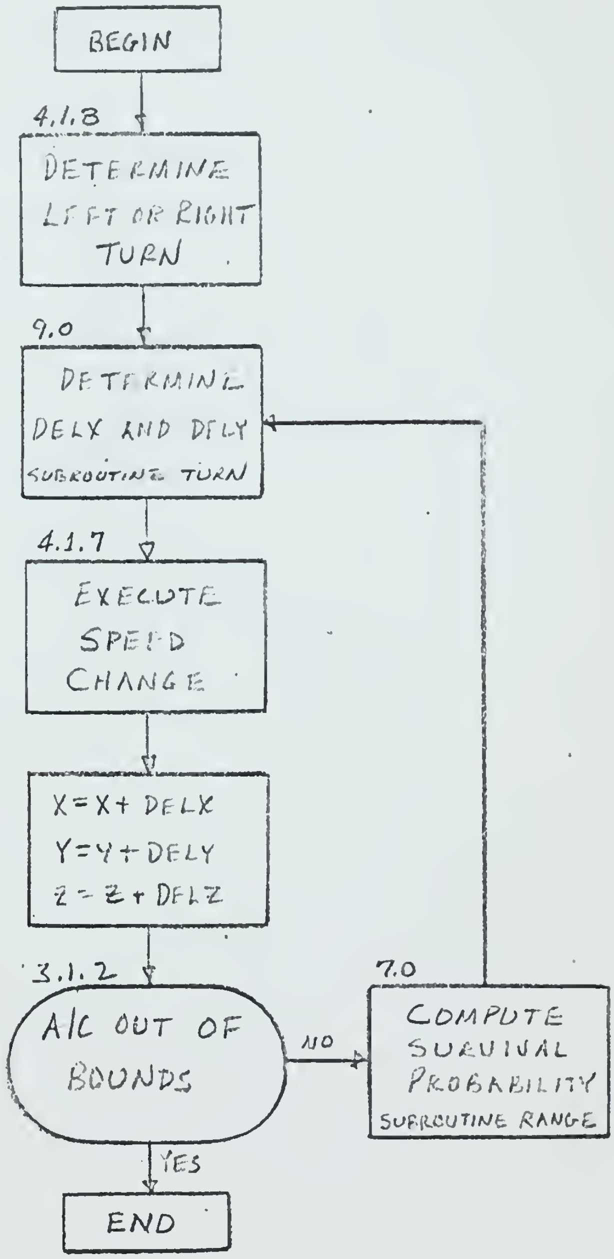



4.1.2
CLIMB STRAIGHT AHEAD
FLOW CHART 4.1.2
PAGE 1 OF 1.
BEGIN
DELZ=AC*DELD
IMARK=IMARK-1
4.1.1
FLY STRAIGHT AND LEVEL
END

4.1.3

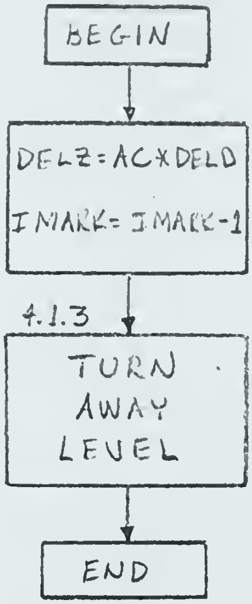
TURN AWAY LEVEL

FLOW CHART 4.1.3
PAGE 1 OF 1



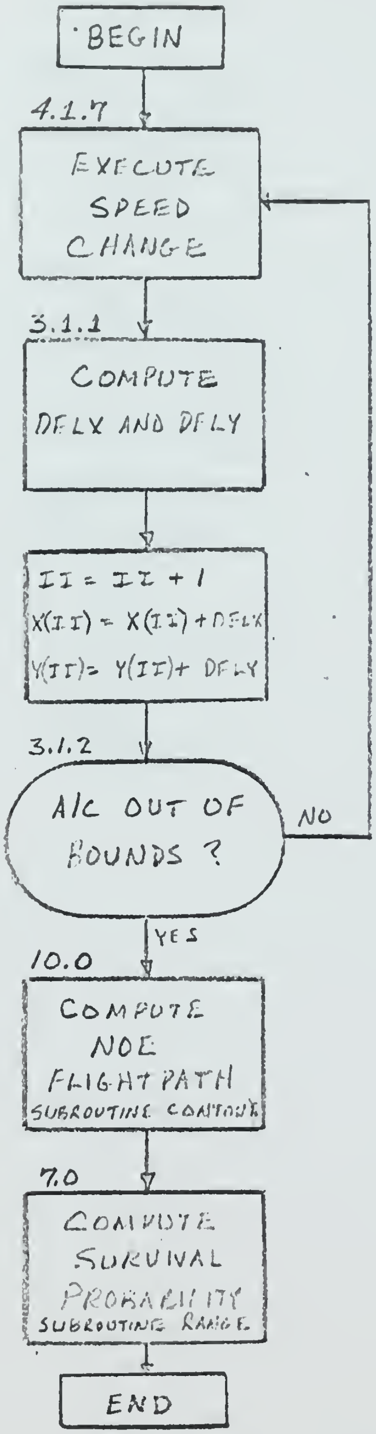
4.1.4

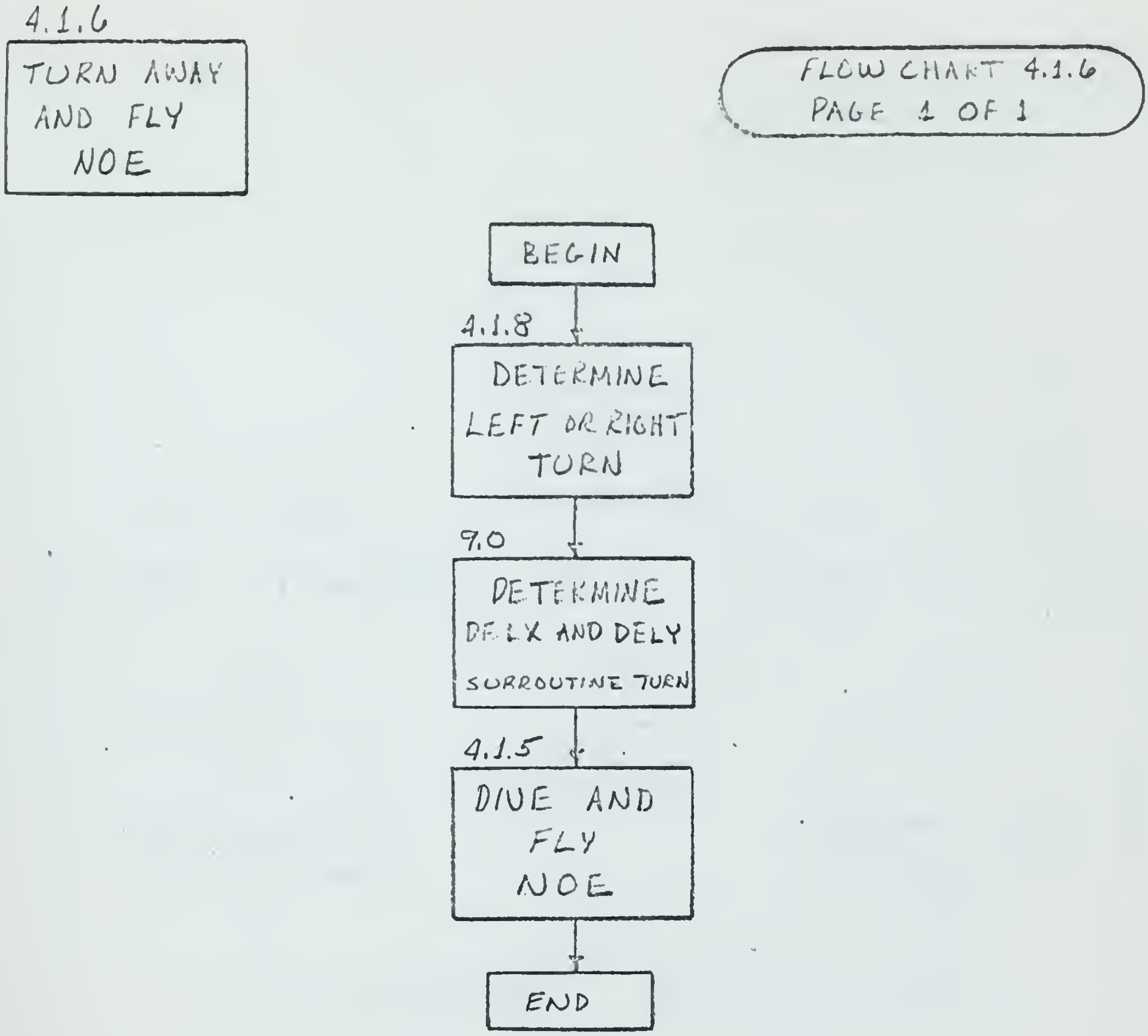
TURN AWAY AND CLIMB



4.1.5

DIVE AND FLY NOE

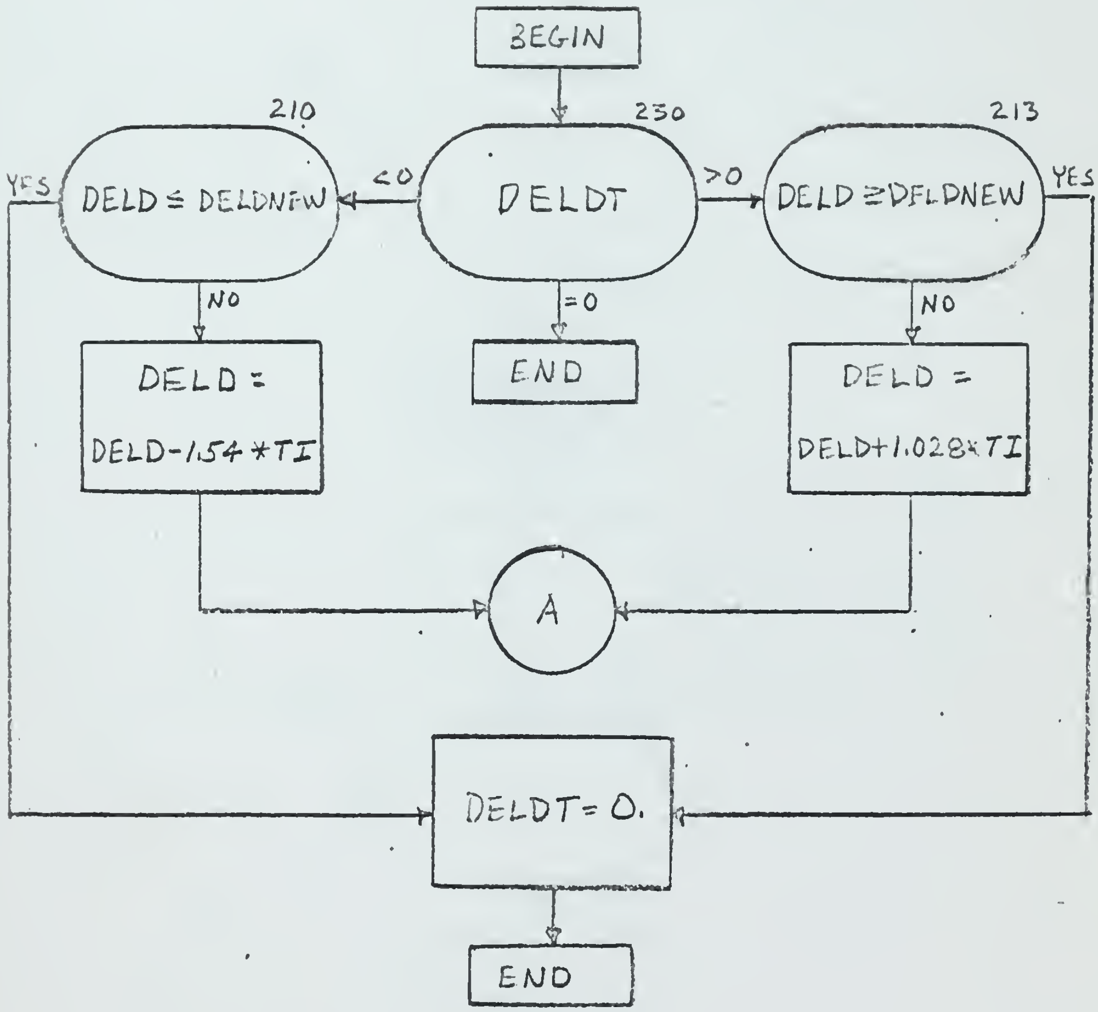


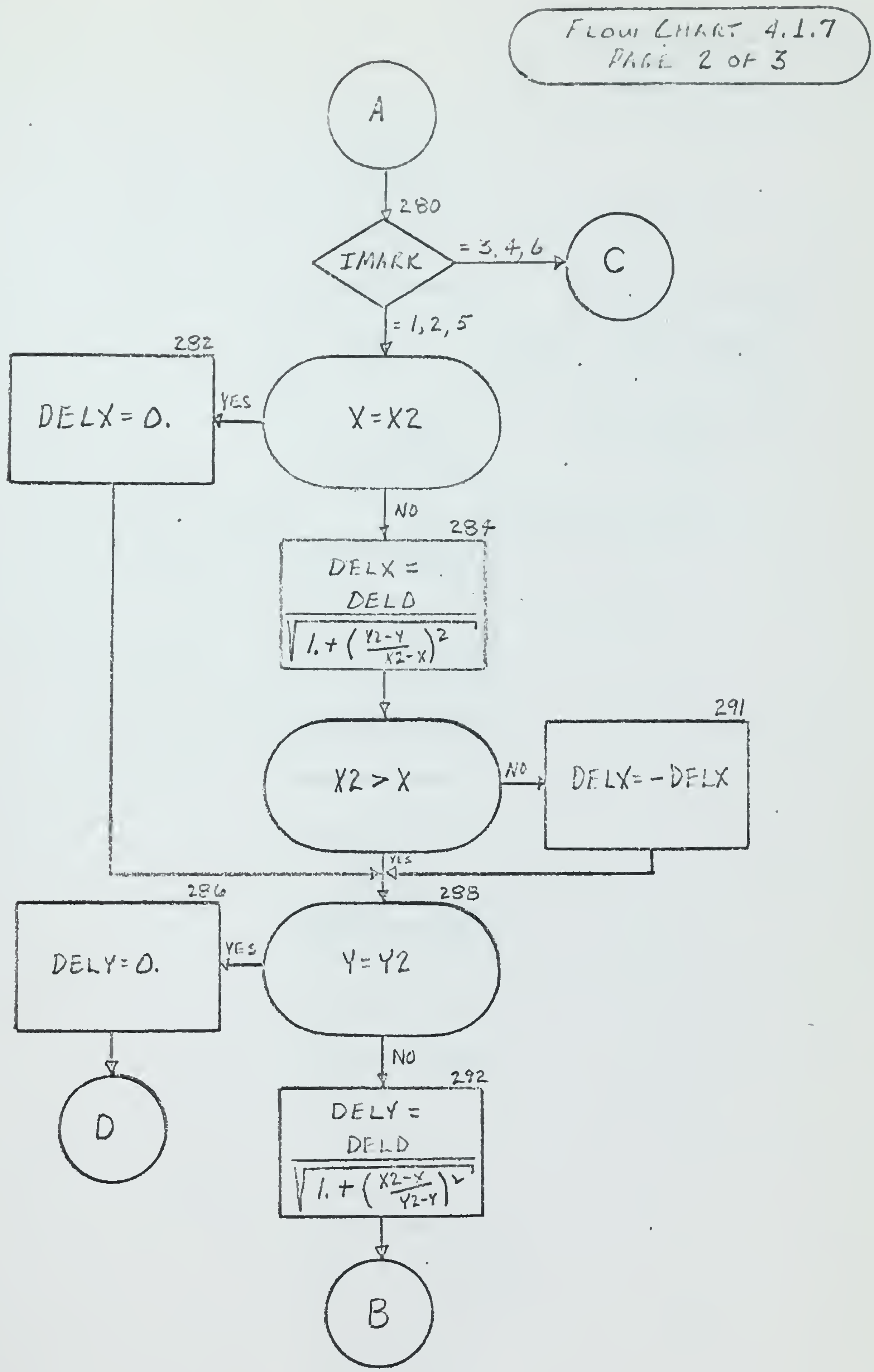

4.1.6
TURN AWAY AND FLY NOE
FLOW CHART 4.1.6
PAGE 1 OF 1
BEGIN
4.1.8
DETERMINE LEFT OR RIGHT TURN
9.0
DETERMINE DELX AND DELY
SUBROUTINE TURN
4.1.5
DIVE AND FLY NOE
END

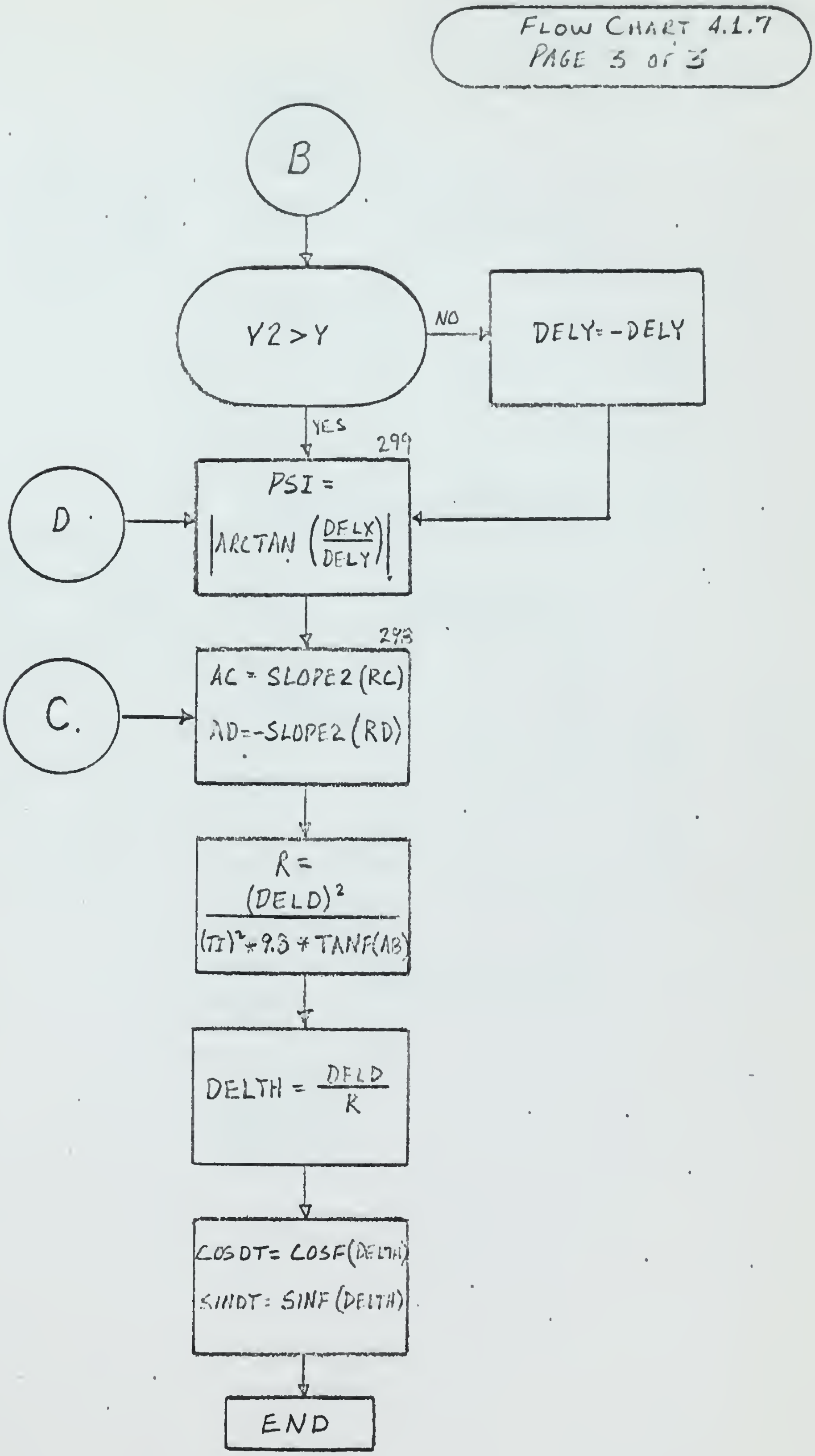
4.1.7

EXECUTE SPEED CHANGE

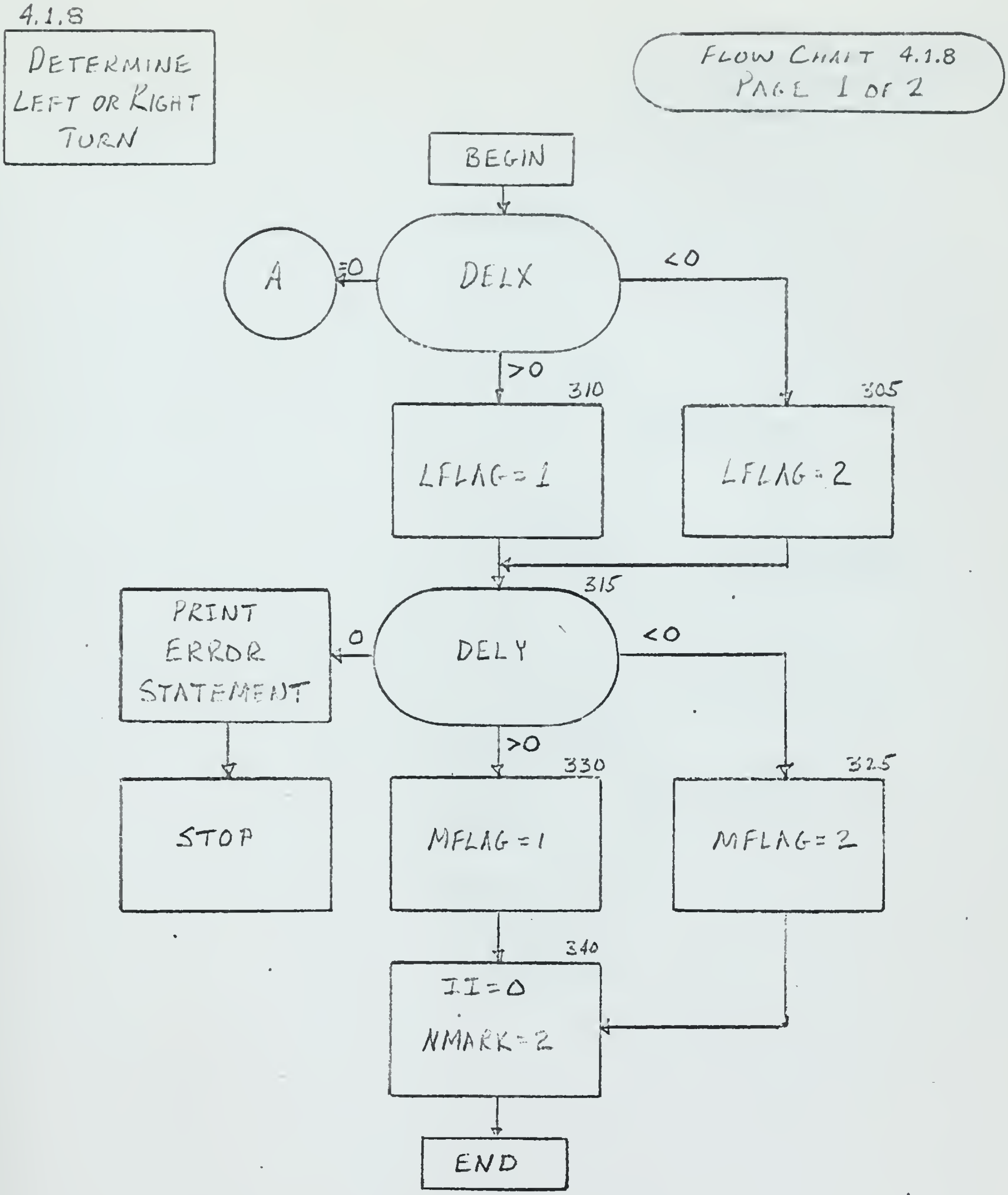
FLOW CHART 4.1.7
PAGE 1 OF 3




FLOW CHART 4.1.7
PAGE 2 OF 3
A
280
IMARK
= 3, 4, 6
C
= 1, 2, 5
282
DELX = 0.
YES
X = X2
NO
284
DELX = DELD / √(1. + ((Y2-Y)/(X2-X))²)
291
X2 > X
NO
DELX = -DELX
YES
286
288
DELY = 0.
YES
Y = Y2
NO
292
DELY = DELD / √(1. + ((X2-X)/(Y2-Y))²)
D
B

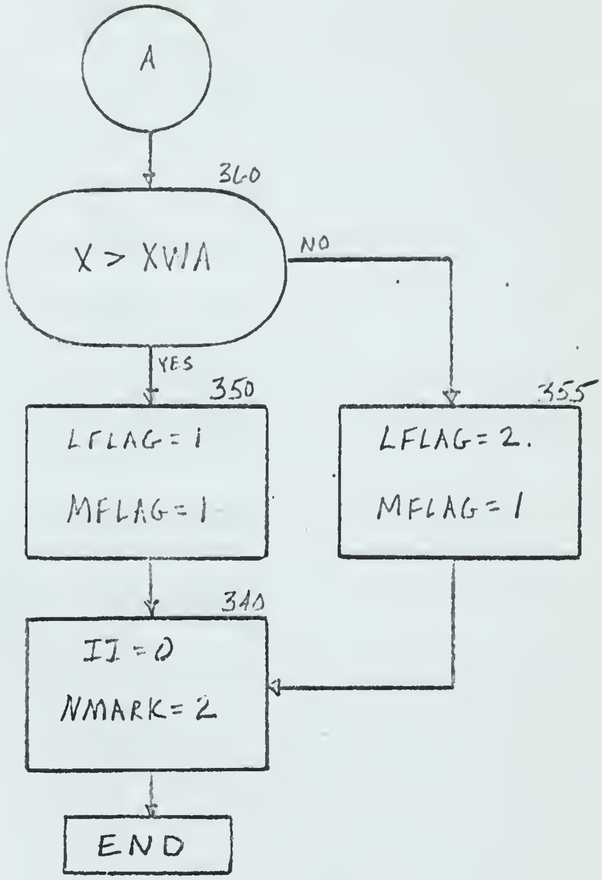


FLOW CHART 4.1.7
PAGE 3 OF 3
B
Y2 > Y
NO
DELY = -DELY
YES
299
D
PSI =
|ARCTAN (DELX/DELY)|
293
C.
AC = SLOPE2 (RC)
AD = -SLOPE2 (RD)
R =
(DELD)²
(TT)² * 9.8 * TANF(AB)
DELTH = DELD/R
COSDT = COSF(DELTH)
SINDT = SINF (DELTH)
END



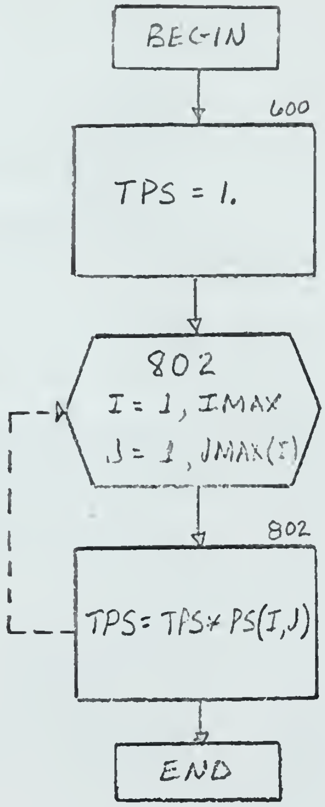
4.1.8
DETERMINE LEFT OR RIGHT TURN
FLOW CHART 4.1.8
PAGE 1 OF 2
BEGIN
A
=0
DELX
<0
>0
310
305
LFLAG=1
LFLAG=2
315
PRINT ERROR STATEMENT
0
DELY
<0
>0
330
325
STOP
MFLAG=1
MFLAG=2
340
II=0
NMARK=2
END

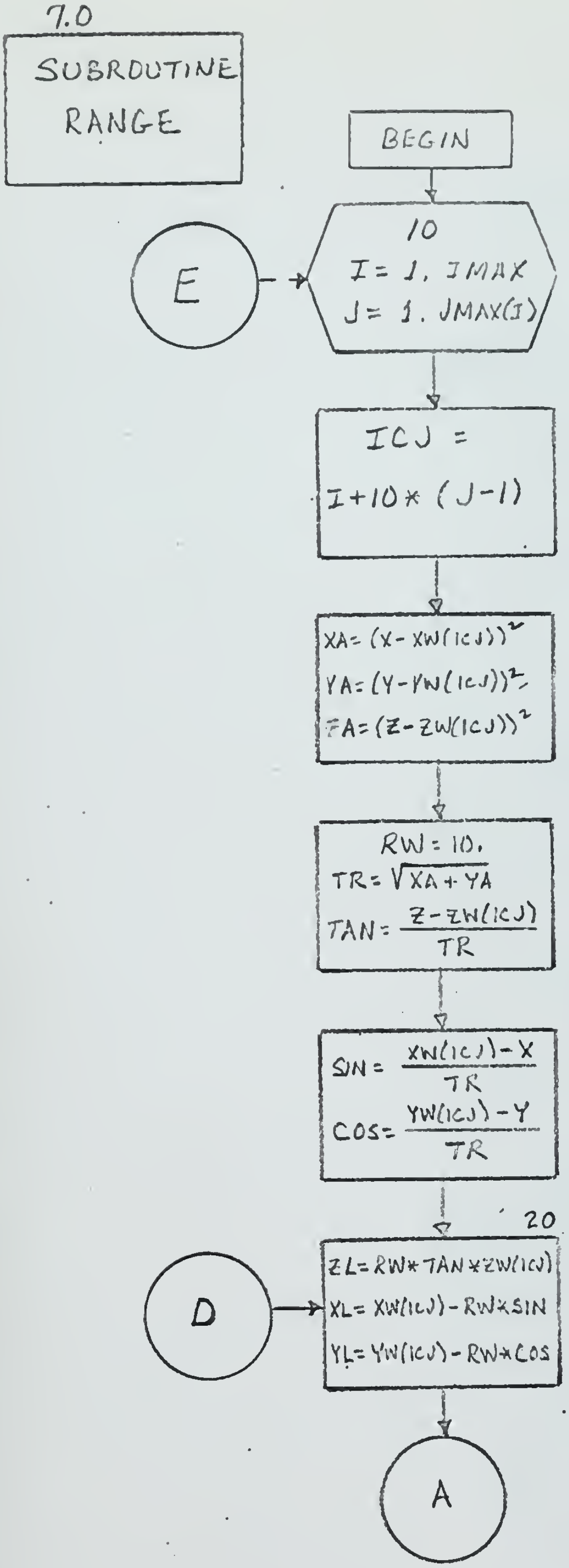
FLOW CHART 4.1.8
PAGE 2 OF 2

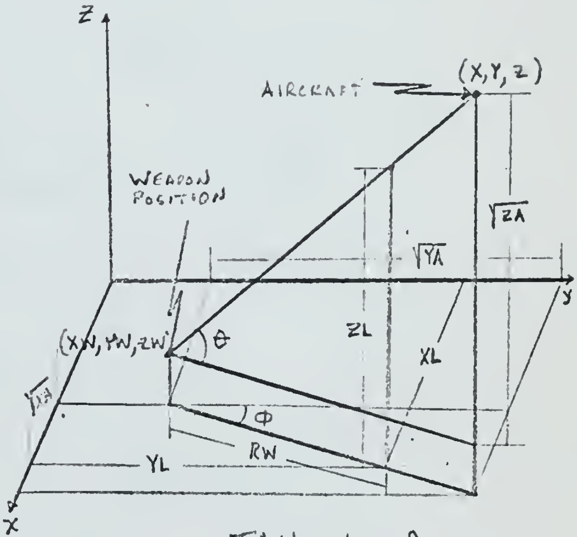


5.1

COMPUTE TOTAL SURVIVAL PROBABILITY



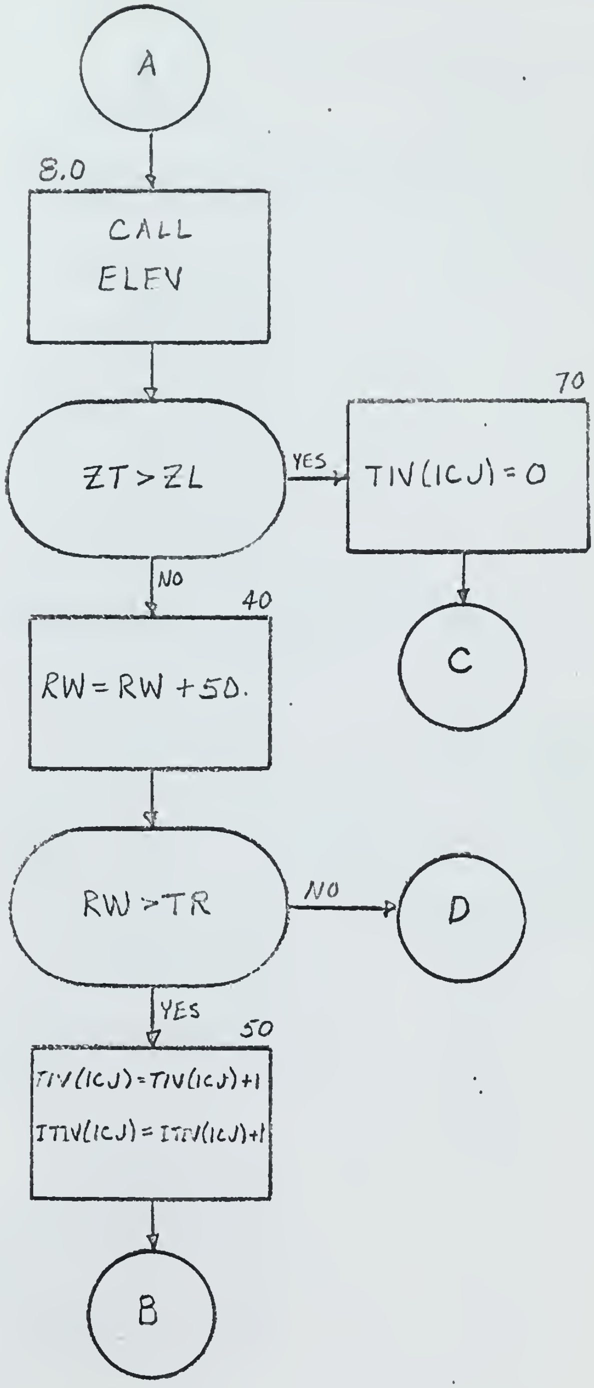


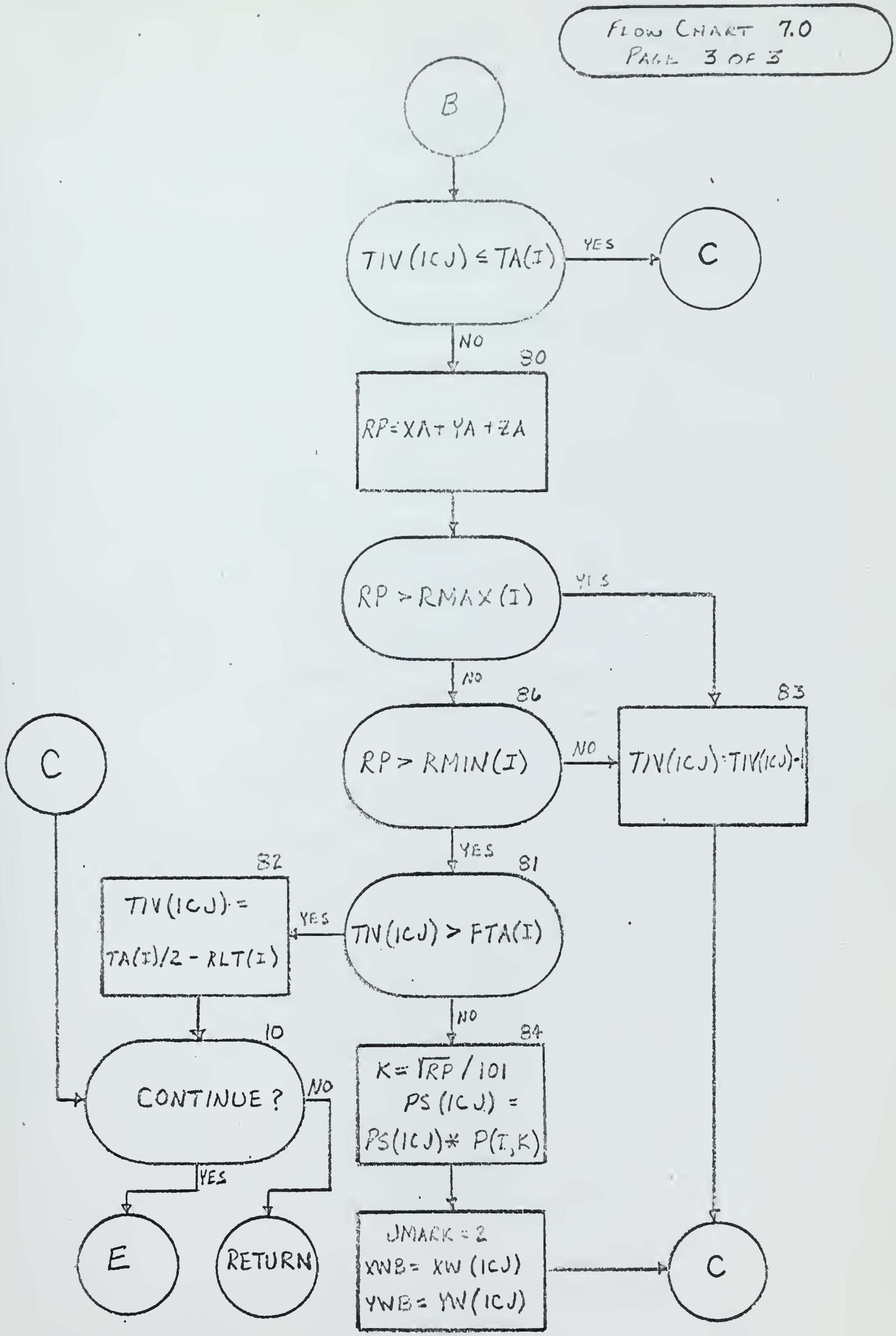


$TAN = \tan\theta$

$SIN = \sin\phi$

$COS = \cos\phi$


A
8.0
CALL
ELEV
ZT > ZL
YES
70
TIV(ICJ) = 0
C
NO
40
RW = RW + 50.
RW > TR
NO
D
YES
50
TIV(ICJ) = TIV(ICJ)+1
ITIV(ICJ) = ITIV(ICJ)+1
B

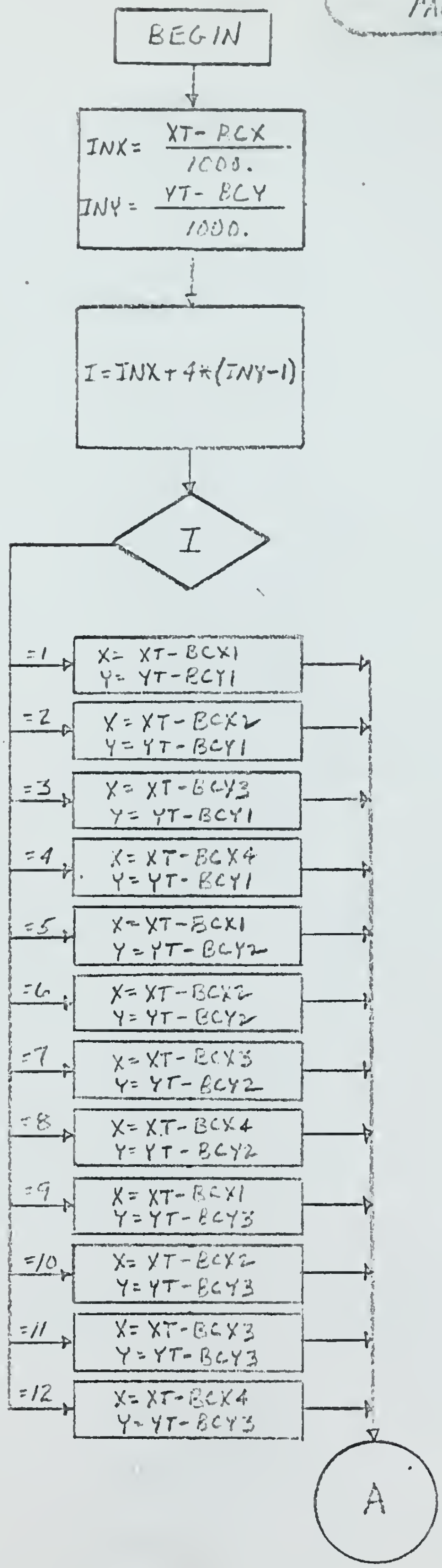


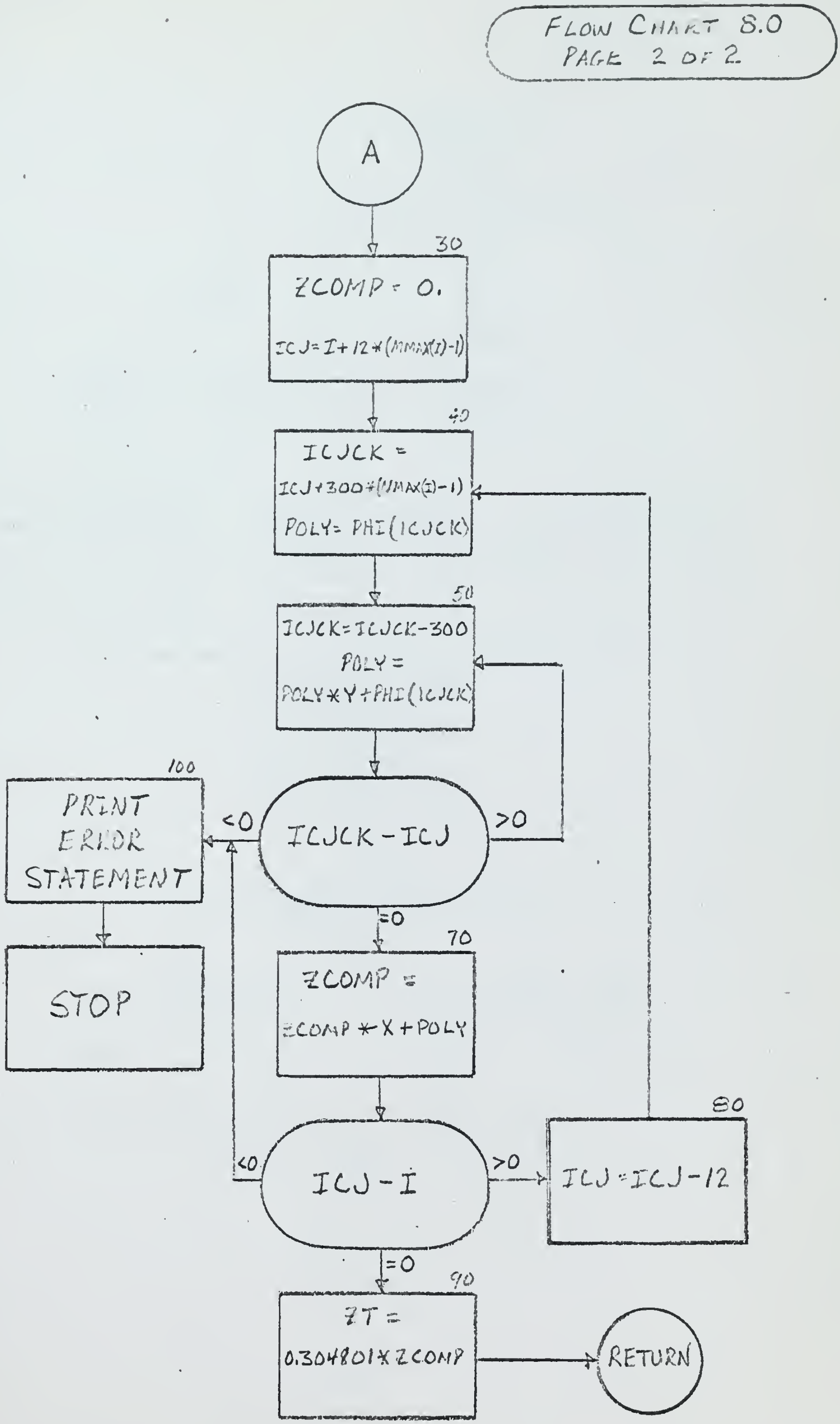
FLOW CHART 7.0
PAGE 3 OF 3
B
TIV(ICJ) ≤ TA(I)
YES
C
NO
80
RP=XA+YA+ZA
RP > RMAX(I)
YES
NO
86
83
RP > RMIN(I)
NO
TIV(ICJ)=TIV(ICJ)-1
C
YES
82
81
TIV(ICJ) =
TA(I)/2 - RLT(I)
YES
TIV(ICJ) > FTA(I)
NO
10
84
CONTINUE ?
NO
K= √RP / 101
PS (ICJ) =
PS(ICJ)* P(I,K)
YES
E
RETURN
JMARK = 2
XWB= XW (ICJ)
YWB= YW (ICJ)
C

8.0

SUBROUTINE ELEV

FLOW CHART 8.0
PAGE 1 OF 2

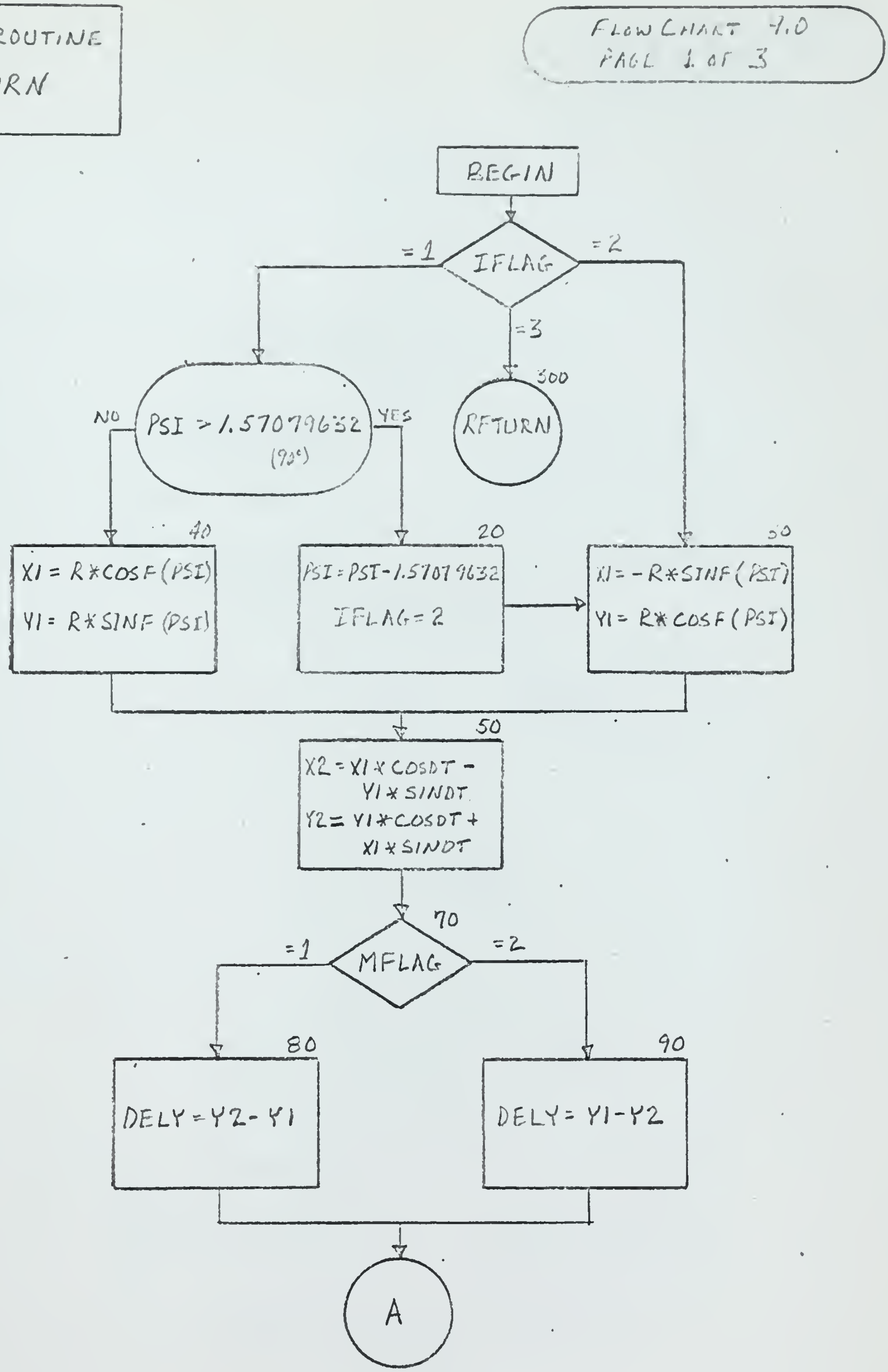


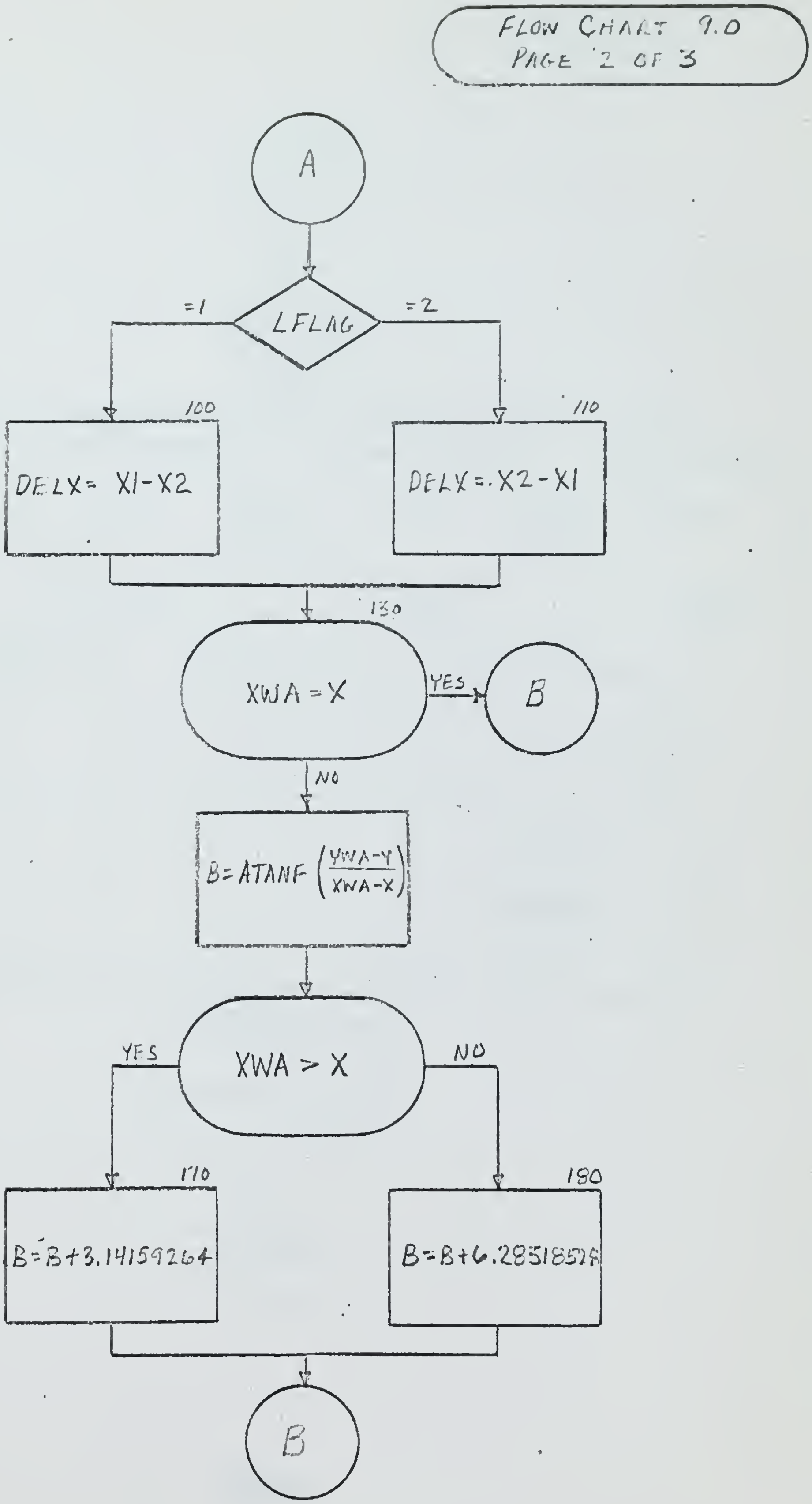


FLOW CHART 8.0
PAGE 2 OF 2
A
30
ZCOMP = 0.
ICJ=I+12*(MMAX(I)-1)
40
ICJCK =
ICJ+300*(NMAX(I)-1)
POLY= PHI(ICJCK)
50
ICJCK=ICJCK-300
POLY =
POLY*Y+PHI(ICJCK)
100
PRINT
ERROR
STATEMENT
<0
ICJCK-ICJ
>0
=0
STOP
70
ZCOMP =
ZCOMP * X + POLY
80
<0
ICJ-I
>0
ICJ=ICJ-12
=0
90
ZT =
0.304801*ZCOMP
RETURN

9.0

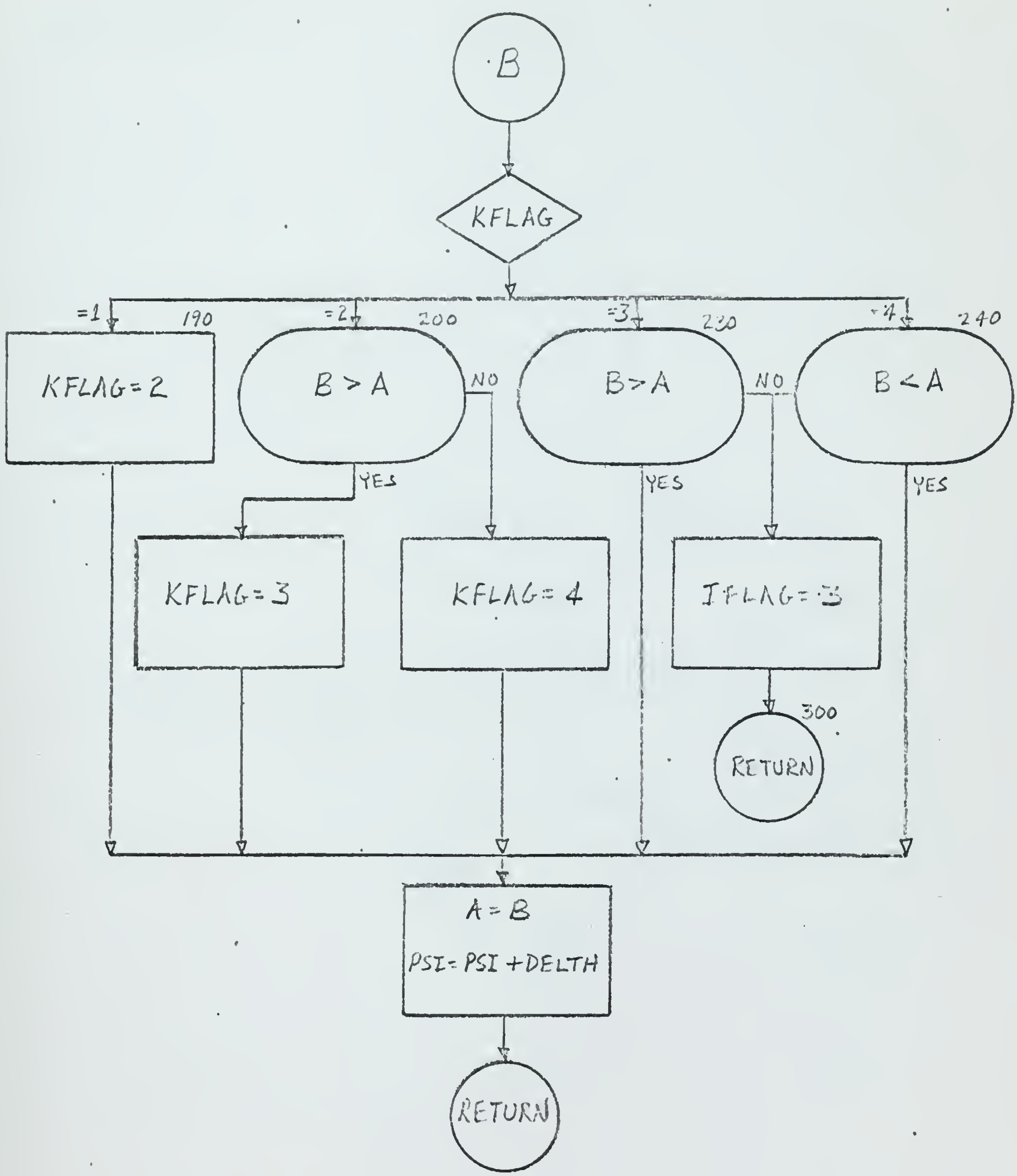
SUBROUTINE TURN

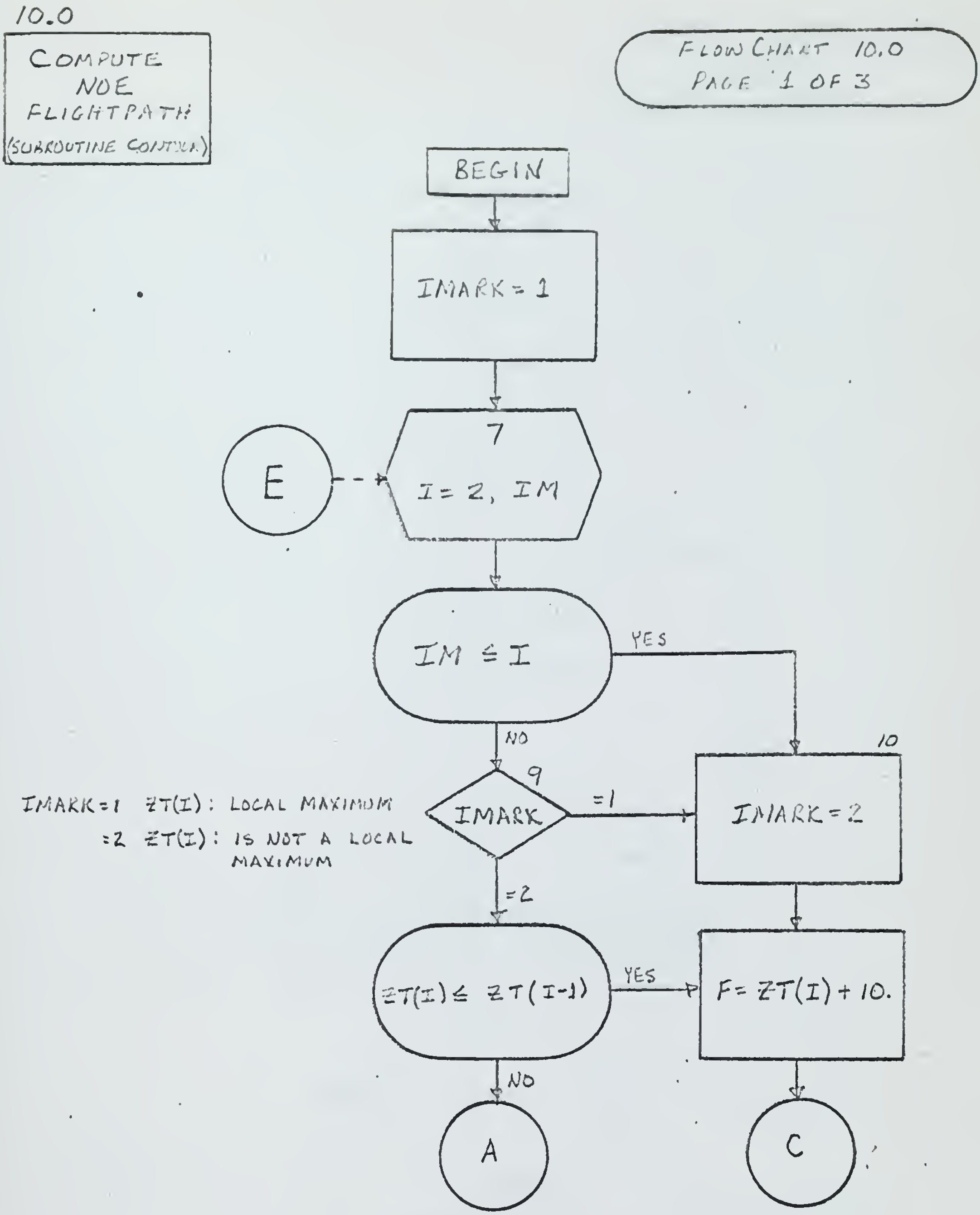




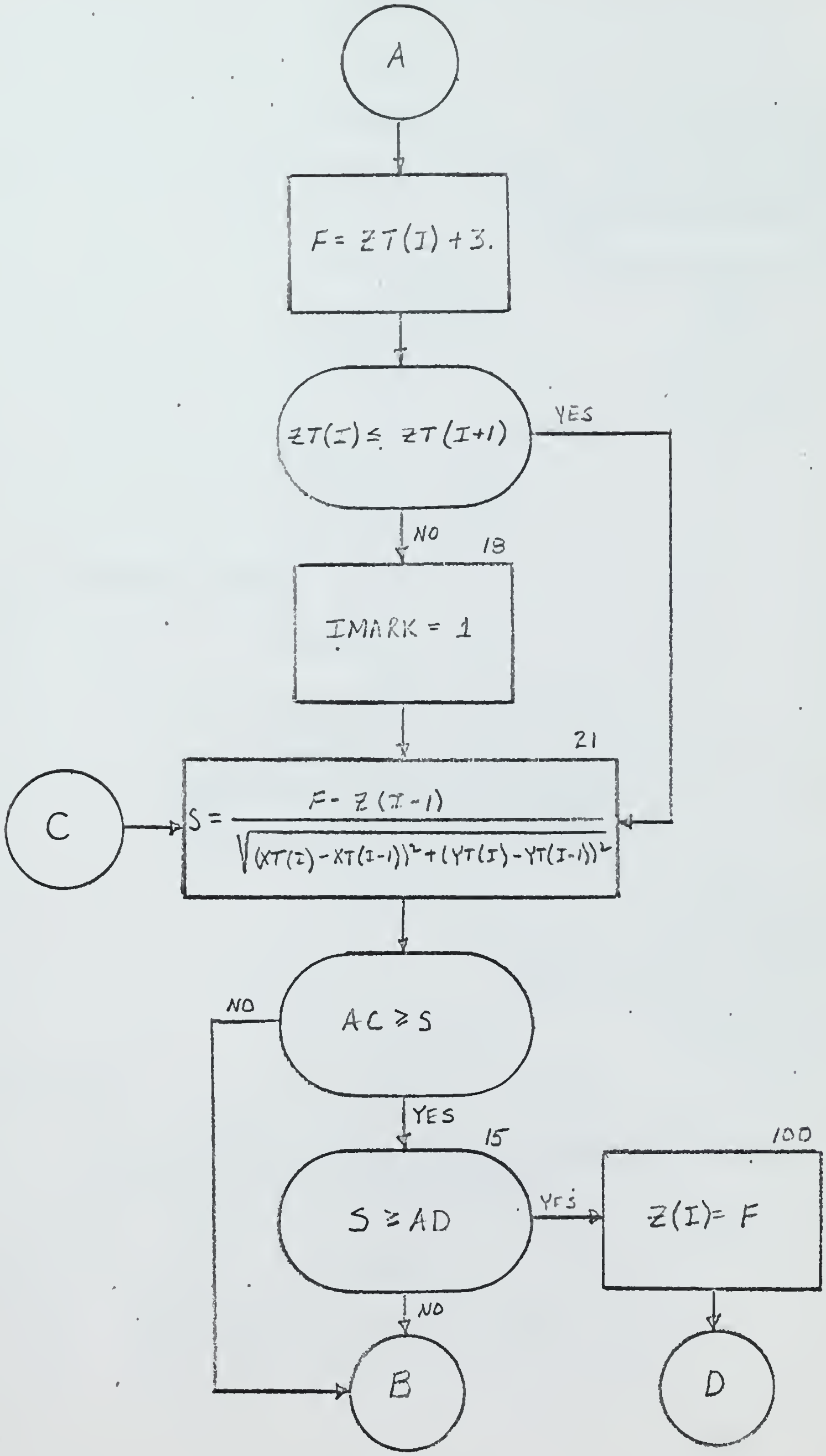
FLOW CHART 9.0
PAGE 2 OF 3
A
LFLAG
=1
=2
100
DELX= X1-X2
110
DELX=.X2-X1
130
XWA = X
YES
B
NO
B=ATANF (YWA-Y / XWA-X)
XWA > X
YES
NO
170
B=B+3.14159264
180
B=B+6.28518528
B

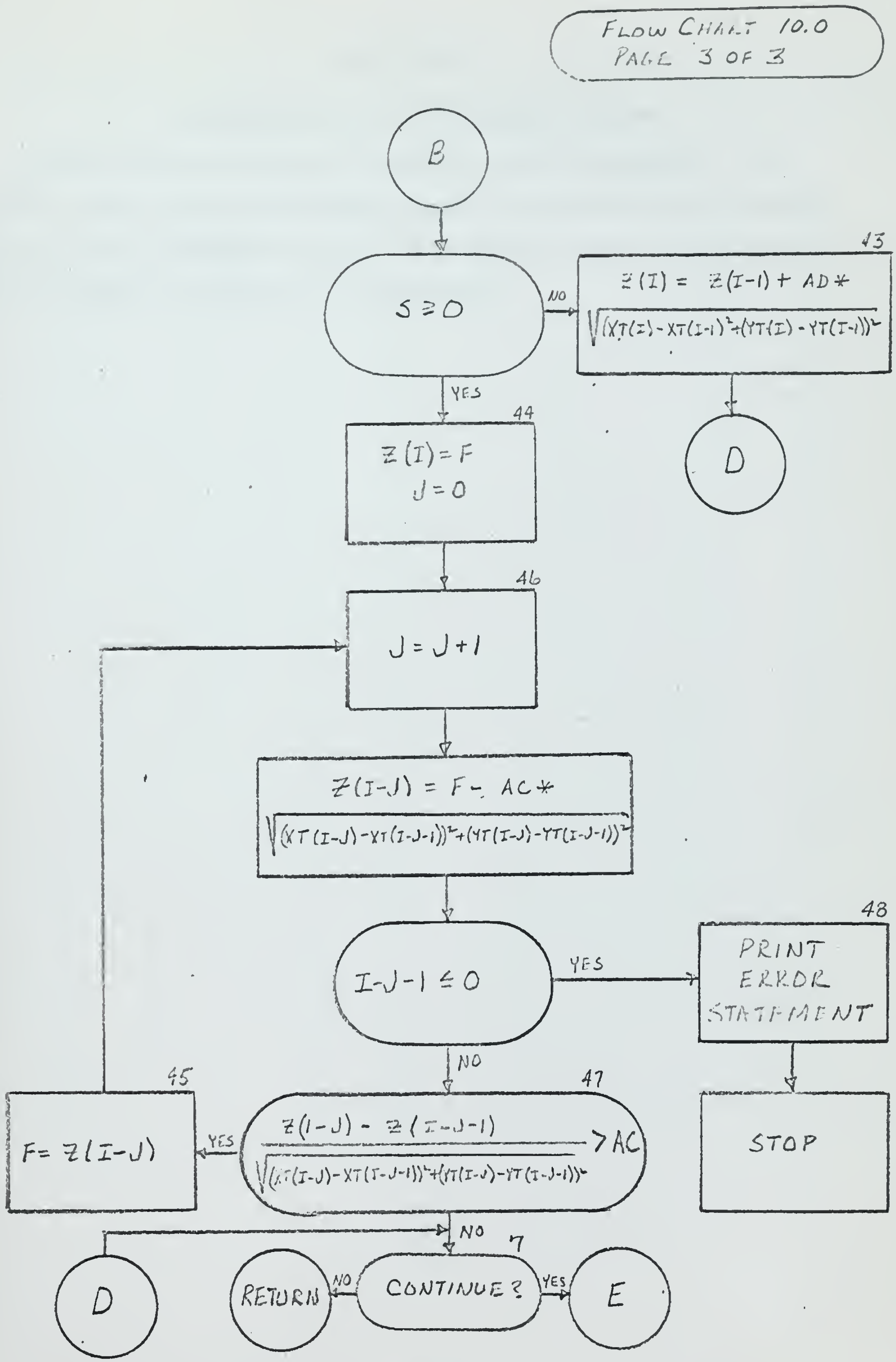
FLOW CHART 9.0
PAGE 3 OF 3.




10.0
COMPUTE
NOE
FLIGHTPATH
(SUBROUTINE CONTOUR)
FLOW CHART 10.0
PAGE 1 OF 3
BEGIN
IMARK = 1
7
I = 2, IM
E
IM ≤ I
YES
NO
10
9
IMARK = 1 ZT(I): LOCAL MAXIMUM
=2 ZT(I): IS NOT A LOCAL MAXIMUM
IMARK
=1
IMARK = 2
=2
ZT(I) ≤ ZT(I-1)
YES
F = ZT(I) + 10.
NO
A
C

FLOW CHART 10.0
PAGE 2 OF 3





FLOW CHART 10.0
PAGE 3 OF 3
B
S ≥ 0
NO
43
Z(I) = Z(I-1) + AD* √((XT(I)-XT(I-1))² + (YT(I)-YT(I-1))²)
YES
44
Z(I) = F
J = 0
D
46
J = J+1
Z(I-J) = F - AC* √((XT(I-J)-XT(I-J-1))² + (YT(I-J)-YT(I-J-1))²)
I-J-1 ≤ 0
YES
48
PRINT ERROR STATEMENT
STOP
NO
45
47
F = Z(I-J)
YES
(Z(I-J) - Z(I-J-1)) / √((XT(I-J)-XT(I-J-1))² + (YT(I-J)-YT(I-J-1))²) > AC
NO
7
D
RETURN
NO
CONTINUE?
YES
E

APPENDIX II

FORTRAN Listing of the PENAIR Program

PENAIR consists of the main program and four subroutines. The variable names used in the PENAIR program are defined in the comments appearing in the FORTRAN listing. The variable names of input and output parameters are explained in Appendix III.

[illegible]

[illegible]

[illegible]

```
      PROGRAM PENAIR                                                    PEN0001
      COMMON/RANG1/XW(10,20),YW(10,20),ZW(10,20),IMAX,JMAX(10),         PEN0002
     A            TIV(10,20),RMAX(10),TA(10),RLT(10),FT(10),RMIN(10),   PEN0003
     B            X,Y,Z,XWA,XWB,YWA,YWB                                 PEN0004
     1      /RANG2/ITIV(10,20),PS(10,20),JMARK,                         PEN0005
     A            P(10,20),FTA(10)                                      PEN0006
     2      /ELEV1/PHI(12,25,25),MMAX(12),NMAX(12),BCX1,BCX2,BCX3,BCX4, PEN0007
     A            BCY1,BCY2,BCY3,XA,YA,ZA,BCX,BCY                       PEN0008
     3      /TURN1/DELTH,R,DELX,DELY,DELD,PSI,IFLAG,KFLAG,LFLAG,MFLAG,  PEN0009
     A            COSDT,SINDT                                           PEN0010
     4      /CONT1/AC,AD,IM,XT(300),YT(300),ZT(300),ZH(300)             PEN0011
      DIMENSION WPNA(10),WPNB(10),NW(10,20),IK(100),AFIELD(16)          PEN0012
      TYPE INTEGER TIV,TA,RLT,FT,FTA                                    PEN0013
C                                                                       PEN0014
      SLOPE(A,B,C)=(A*B)/(60.*C) $ SLOPE2(A)=.00508*TI*A/DELD           PEN0015
C     COMPUTES THE SLOPE OF THE MAXIMUM CLIMB AND DESCENT OF THE A/C.   PEN0016
C                                                                       PEN0017
C                                                                       PEN0018
      READ 1010,NRUN                                                    PEN0019
      READ 1010,KMAX,XMIN,YMIN,XMAX,YMAX,TI,TPD                         PEN0020
 1010 FORMAT(I10,6F10.0)                                                PEN0021
C     KMAX=NUMBER OF POLYNOMIALS TO APPROXIMATE THE TERRAIN.            PEN0022
C     XMIN=X COORDINATE OF LEFT EDGE OF THE TERRAIN.                    PEN0023
C     YMIN=Y COORDINATE OF BOTTOM EDGE                                  PEN0024
C     XMAX=X COORDINATE OF RIGHT EDGE                                   PEN0025
C     YMAX=Y COORDINATE OF TOP EDGE                                     PEN0026
C     TI=GAME TIME INTERVAL (SECONDS)                                   PEN0027
C     TPD=DIMENSION OF TERRAIN SQUARE APPROXIMATED BY ONE POLYNOMIAL.   PEN0028
C                                                                       PEN0029
      IF(KMAX.GT.12)9,10                                                PEN0030
    9 PRINT 1008                                                        PEN0031
 1008 FORMAT(1H1////77H NUMBER OF APPROXIMATING TERRAIN POLYNOMIALS EXCEPEN0032
     *EDS 12. THIS RUN TERMINATED.    )                                 PEN0033
      STOP                                                              PEN0034
   10 READ 1090,X,Y,Z,X2,Y2                                             PEN0035
 1090 FORMAT(5F10.0)                                                    PEN0036
```

```
C     X,Y,Z ARE THE BEGINNING COORDINATES OF THE AIRCRAFT.                  PEN0037
C     X2,Y2 ARE THE COORDINATES OF THE DESIRED DESTINATION OF THE A/C.      PEN0038
C                                                                           PEN0039
   20 READ 1090,GMAX,RC,RD,GS                                               PEN0040
C     GMAX=MAXIMUM POSITIVE G TO BE USED BY THE AIRCRAFT.                   PEN0041
C     RC=MAXIMUM RATE OF CLIMB (FT./MIN)                                    PEN0042
C     RD=MAXIMUM RATE OF DESCENT (FT./MIN)                                  PEN0043
C     GS=ESTIMATED GROUND SPEED (KNOTS)                                     PEN0044
C                                                                           PEN0045
      READ 1010,IMAX                                                        PEN0046
C     IMAX=THE NUMBER OF TYPES OF WEAPONS                                   PEN0047
C                                                                           PEN0048
      IF(IMAX.GT.10)15,16                                                   PEN0049
   15 PRINT 1015                                                            PEN0050
 1015 FORMAT(1H1////56H WEAPON DIMENSIONS ARE EXCEEDED. THIS RUN IS TERMPEN0051
     *INATED.  )                                                            PEN0052
      STOP                                                                  PEN0053
   16 DO 40 I=1,IMAX                                                        PEN0054
C                                                                           PEN0055
      READ 1030, WPNA(I),WPNB(I),JMAX(I),TA(I),FT(I),RLT(I),RMAX(I),        PEN0056
     1RMIN(I)                                                               PEN0057
 1030  FORMAT(2A8,4X,4I5,2F10.0)                                            PEN0058
C     WPNA,WPNB=TYPE OF WEAPON (COL.1-16)                                   PEN0059
C     JMAX(I)=NUMBER OF WEAPONS OF TYPE I                                   PEN0060
C     TA=TIME REQUIRED TO ACQUIRE TARGET (SEC.)                             PEN0061
C     FT=FIRING TIME BEFORE RELOADING (SEC.)                                PEN0062
C     RLT=TIME REQUIRED TO RELOAD (SEC.)                                    PEN0063
C     RMAX=MAXIMUM RANGE OF WEAPON (METERS)                                 PEN0064
C     RMIN=MINIMUM RANGE OF WEAPON (METERS)                                 PEN0065
C                                                                           PEN0066
      IF(JMAX(I).GT.20)15,17                                                PEN0067
   17 READ 1040, (P(I,K),K=1,20)                                            PEN0068
 1040 FORMAT(10F8.0)                                                        PEN0069
C     P(I,K)=PROBABILITY OF A KILL BY WEAPON I AT SLANT RANGE K             PEN0070
C                                                                           PEN0071
      JMAXX=JMAX(I)                                                         PEN0072
```

```
C                                                                       PEN0073
   40 READ 1050,(NW(I,J),XW(I,J),YW(I,J),ZW(I,J),J=1,JMAXX)             PEN0074
 1050 FORMAT(I2,8X,3F10.0)                                              PEN0075
C     NW=NUMBER ASSIGNED TO THE WEAPON                                  PEN0076
C     XW,YW=X AND Y COORDINATES OF THE WEAPON                           PEN0077
C                                                                       PEN0078
      DO 96 I=1,KMAX                                                    PEN0079
C                                                                       PEN0080
      READ 1060, MMAX(I),NMAX(I)                                        PEN0081
 1060 FORMAT(2I10)                                                      PEN0082
C     MMAX=DEGREE OF X IN THE ITH TERRAIN POLYNOMIAL                    PEN0083
C     NMAX=DEGREE OF Y                                                  PEN0084
      MMAXX=MMAX(I)=MMAX(I)+1 $ NMAXX=NMAX(I)=NMAX(I)+1                 PEN0085
C                                                                       PEN0086
      IF(MMAX(I).GT.25.OR.NMAX(I).GT.25)95,96                           PEN0087
   95 PRINT 1095                                                        PEN0088
 1095 FORMAT(1H1////78H MAXIMUM DEGREE OF THE TERRAIN POLYNOMIAL IS EXCEPEN0089
     *EDED. THIS RUN IS TERMINATED.  )                                  PEN0090
      STOP                                                              PEN0091
   96 READ 1070,((PHI(I,K,L),L=1,NMAXX),K=1,MMAXX)                      PEN0092
 1070 FORMAT(4E20.10)                                                   PEN0093
C     PHI(I,K,L)=COEFFICIENTS OF TERRAIN POLYNOMIAL                     PEN0094
C       I=ITH TERRAIN POLYNOMIAL                                        PEN0095
C       K=DEGREE OF X                                                   PEN0096
C       L=DEGREE OF Y                                                   PEN0097
C                                                                       PEN0098
      BCX=XMIN-TPD $ BCY=YMIN-TPD                                       PEN0099
      BCX1 = XMIN+TPD/2. $ BCY1 = YMIN+TPD/2.                           PEN0100
      BCX2=BCX1+TPD $ BCX3=BCX2+TPD $ BCX4=BCX3+TPD                     PEN0101
      BCY2=BCY1+TPD $ BCY3=BCY2+TPD                                     PEN0102
      PRINT 2025                                                        PEN0103
 2025 FORMAT (1H1///12X,22HSUMMARY OF WEAPON DATA  //)                  PEN0104
      PRINT 2020,(WPNA(I),WPNB(I),I=1,IMAX)                             PEN0105
 2020 FORMAT(1H0,   11HWEAPON TYPE, 10X, 12A8)                          PEN0106
      PRINT 2024,(JMAX(I),I=1,IMAX)                                     PEN0107
 2024 FORMAT( 1X, 17HNUMBER OF WEAPONS,    I10,5I16)                    PEN0108
```

```
      PRINT 2026,(TA(I),I=1,IMAX)
 2026 FORMAT( 1X, 22HACQUISITION TIME (SEC), I5,5I16)
      PRINT 2028,(FT(I),I=1,IMAX)
 2028 FORMAT( 1X, 17HFIRING TIME (SEC),    I10,5I16)
      PRINT 2030,(RLT(I),I=1,IMAX)
 2030 FORMAT( 1X, 20HRELOADING TIME (SEC),I7,5I16)
      PRINT 2032,(RMIN(I),I=1,IMAX)
 2032 FORMAT( 1X, 22HMINIMUM RANGE (METERS), F6.0,5F16.0)
      PRINT 2034,(RMAX(I),I=1,IMAX)
 2034 FORMAT( 1X, 22HMAXIMUM RANGE (METERS),F6.0,5F16.0)
      PRINT 2036
 2036 FORMAT(1H0, 6X, 6HWEAPON, 9X, 4HSITE, 6X, 6HWEAPON /
     1            8X  4HTYPE, 9X, 6HNUMBER, 5X, 9HPOSITIONS /)
      DO 180 I=1,IMAX
C
C     CONVERT KILL PROBABILITIES TO SURVIVAL PROBABILITIES.
      DO 30 K=1,20
   30 P(I,K)=1.-P(I,K)
C
      TA(I)=TA(I)/TI
      FT(I)=FT(I)/TI
      FTA(I)=TA(I)+FT(I)
      RLT(I)=RLT(I)/TI
      RMAX(I)=RMAX(I)**2 $ RMIN(I)=RMIN(I)**2
      PRINT 2038,WPNA(I),WPNB(I)
 2038 FORMAT( 5X,2A8)
      JMAXX=JMAX(I)
      DO 180 J=1,JMAXX $ ICJ=I+10*(J-1)
      XA=XW(ICJ) $ YA=YW(ICJ) $ CALL ELEV
C
C     ASSIGN THE HIGHER OF THE ACTUAL OR COMPUTED ELEVATION TO THE WPN.
      IF(ZA.GT.ZW(ICJ))50,180
   50 ZW(ICJ) = ZA
C
  180 PRINT 2040, NW(ICJ),XW(ICJ),YW(ICJ),ZW(ICJ)
 2040 FORMAT(22X,I3,5X,2F7.0,F5.0)
```

```
      PRINT 2041
 2041 FORMAT (1H1)
C
C     COMPUTE THE MAXIMUM ANGLE OF BANK OF THE A/C.
      AB=ACOSF(1./GMAX)
C
      XX=X $ YY=Y $ ZZ=Z
      DO 9990 N=1,NRUN
      READ 1000,(AFIELD(I),I=1,16)
 1000 FORMAT(4A8)
C     THIS READS IN THE TITLE OF THE RUN
C
      READ 1075,IMARK,IPUNCH,GSNEW
 1075 FORMAT(2I10,F10.0)
C     IMARK=1  FLIGHTPATH UPON RECEIVING FIRE  STRAIGHT AND LEVEL
C           2                                  CLIMB STRAIGHT AHEAD
C           3                                  TURN AWAY, LEVEL
C           4                                  TURN AWAY AND CLIMB
C           5                                  DIVE AND FLY NOE
C           6                                  TURN AWAY AND FLY NOE
C           7                COMPLETELY PREPLANNED
C           8                X AND Y COORDINATES PREPLANNED,NOE COMPUTED
C     IPUNCH=1 A/C FLIGHT PATH WILL BE PUNCHED ON CARDS.
C           =0 A/C FLIGHT PATH WILL NOT BE PUNCHED ON CARDS.
C     GSNEW=GROUND SPEED AFTER RECEIVING FIRE (KNOTS)
      IF(IPUNCH)182,183,182
  182 PUNCH 1001,(AFIELD(I),I=1,4)
 1001 FORMAT (4A8)
  183 PRINT 2000,(AFIELD(I),I=1,4)
 2000 FORMAT(////31X,10HSUMMARY OF/28X,4A8)
      PRINT 2010,(AFIELD(I),I=5,16)
 2010 FORMAT (19X,10HMISSION - ,4A8/(29X,4A8))
      DO 181 I=1,IMAX $ JMAXX=JMAX(I)
      DO 181 J=1,JMAXX
      ITIV(I,J)=TIV(I,J)=0
  181 PS(I,J)=1.
```

```
      JMARK=KMARK=LMARK=NMARK=IFLAG=KFLAG=1 $ DELZ=0.
      X=XX $ Y=YY $ Z=ZZ
      IF(IMARK.LE.6) 220,100
C
C     COMPUTE THE DISTANCE (METERS) TRAVELED BY THE A/C IN ONE TIME
C     INTERVAL.
  220 DELD=0.5144444444*TI*GS
C
C     COMPUTE THE SLOPE OF THE MAXIMUM CLIMB AND DESCENT OF THE A/C.
      AC=SLOPE(RC,TI,GS) $ AD=-SLOPE(RD,TI,GS)
C     COMPUTE THE SPEED TO BE FLOWN UPON RECEIVING FIRE.
      DELDNEW=0.5144444444*TI*GSNEW $ DELDT=DELDNEW-DELD $ GO TO 281
C
  100 I=0
  110 I=I+1
C
      IF(I.GT.300)111,112
  111 PRINT 1111
 1111 FORMAT(1H1////72H PREPLANNED FLIGHTPATH INPUT POINTS EXCEEDS 300.
     *THIS RUN IS TERMINATED.  )
      STOP
  112 READ 1080,NPATH,XT(I),YT(I),ZH(I)
 1080 FORMAT(5X,I2,10X,3F9.1)
C     NPATH=PREPLANNED FLIGHT PATH NUMBER (ANY POSITIVE INTEGER )
C     XG(I)=PREPLANNED X COORDINATE IN THE ITH TIME INTERVAL
C     YG(I)=PREPLANNED Y COORDINATE
C     ZH(I)=PREPLANNED Z COORDINATE
C
      IF(NPATH)110,120,110
  120 IM=I-1
      DO 130 I=1,IM
      XA=XT(I) $ YA=YT(I) $ CALL ELEV
  130 ZT(I)=ZA
      IMARK=IMARK-6 $ GO TO (540,260)IMARK
  260 AC=SLOPE(RC,TI,GS) $ AD=-SLOPE(RD,TI,GS) $ GO TO 530
  230 IF(DELDT)210,234,213
```

```
  210 IF(DELD.LE.DELDNEW)214,215                                  PEN0217
  214 DELDT=0. $ GO TO 234                                        PEN0218
  213 IF(DELD.GE.DELDNEW)214,216                                  PEN0219
  215 DELD=DELD-1.5444444444*TI $ GO TO 280                       PEN0220
  216 DELD=DELD+1.0288888888*TI                                   PEN0221
  280 GO TO(281,281,298,298,281,298)IMARK                         PEN0222
  281 IF(X.EQ.X2)282,284                                          PEN0223
  282 DELX=0. $ GO TO 288                                         PEN0224
  284 DELX = DELD/SQRTF(1.+((Y2-Y)/(X2-X))**2)                    PEN0225
      IF(X2.GT.X)288,291                                          PEN0226
  291 DELX = -DELX                                                PEN0227
  288 IF(Y.EQ.Y2)286,292                                          PEN0228
  286 DELY=0.                                                     PEN0229
      GO TO 299                                                   PEN0230
  292 DELY = DELD/SQRTF(1.+((X2-X)/(Y2-Y))**2)                    PEN0231
      IF(Y2.GT.Y)299,293                                          PEN0232
  293 DELY =-DELY                                                 PEN0233
  299 PSI=ABSF(ATANF(DELX/DELY))                                  PEN0234
  298 AC=SLOPE2(RC) $ AD=-SLOPE2(RD)                              PEN0235
C                                                                 PEN0236
C     COMPUTE THE RADIUS OF TURN OF THE A/C (METERS).             PEN0237
      R=(DELD**2)/(TI**2*9.80616*TANF(AB)) $ DELTH=DELD/R         PEN0238
C                                                                 PEN0239
      COSDT=COSF(DELTH) $ SINDT=SINF(DELTH)                       PEN0240
  234 X=X+DELX $ Y=Y+DELY $ Z=Z+DELZ                              PEN0241
      IF(X.LT.XMIN)238,231                                        PEN0242
  231 IF(X.GT.XMAX)238,232                                        PEN0243
  232 IF(Y.LT.YMIN)238,233                                        PEN0244
  233 IF(Y.GT.YMAX)238,235                                        PEN0245
  238 GO TO (600,520,520)LMARK                                    PEN0246
  235 IF(IPUNCH)237,236,237                                       PEN0247
  237 XA=X $ YA=Y $ CALL ELEV                                     PEN0248
      PUNCH 4000,X,Y,Z,ZA                                         PEN0249
 4000 FORMAT (6X,1H1,10X,4F9.1)                                   PEN0250
  236 GO TO (239,370,510)LMARK                                    PEN0251
  239 CALL RANGE                                                  PEN0252
```

```
  240 GO TO (234,245)JMARK                                                  PEN0253
C                                                                           PEN0254
C     JMARK=1 IF NO WEAPONS HAVE FIRED.                                     PEN0255
C          =2 IF ANY WEAPON HAS FIRED.                                      PEN0256
C     IF JMARK=2, THEN                                                      PEN0257
C     LMARK=1 IF IMARK=1,2,3,4                                              PEN0258
C          =2 IF      =6                                                    PEN0259
C          =3 IF      =5                                                    PEN0260
C                                                                           PEN0261
  245 GO TO(230,250,301,250,500,302)IMARK                                   PEN0262
  250 DELZ=AC*DELD $ IMARK=IMARK-1 $ GO TO 230                              PEN0263
  302 LMARK=2                                                               PEN0264
  301 GO TO (303,370)NMARK                                                  PEN0265
C                                                                           PEN0266
C     NMARK=1 BEFORE TURN DIRECTION HAS BEEN DETERMINED.                    PEN0267
C          =2 AFTER  TURN DIRECTION HAS BEEN DETERMINED.                    PEN0268
C                                                                           PEN0269
  303 XWA=XWB $ YWA=YWB                                                     PEN0270
      IF(DELX)305,360,310                                                   PEN0271
  305 LFLAG=2 $ GO TO 315                                                   PEN0272
  310 LFLAG=1                                                               PEN0273
  315 IF(DELY)325,320,330                                                   PEN0274
  325 MFLAG=2 $ GO TO 340                                                   PEN0275
  330 MFLAG=1 $ GO TO 340                                                   PEN0276
  320 PRINT 7000                                                            PEN0277
 7000 FORMAT(1H1,10X,53HILLEGAL INPUT, DELY EQUALS ZERO, THIS RUN TERMINPEN0278
     1ATED.)                                                                PEN0279
      STOP                                                                  PEN0280
  360 IF(X.GT.XWA)350,355                                                   PEN0281
  350 LFLAG=1 $ MFLAG=1 $ GO TO 340                                         PEN0282
  355 LFLAG=2 $ MFLAG=1                                                     PEN0283
  340 II=0 $ NMARK=2                                                        PEN0284
  370 CALL TURN                                                             PEN0285
      GO TO (230,510)LMARK                                                  PEN0286
  500 II=0 $ LMARK=3                                                        PEN0287
  510 II=II+1 $ XT(II)=XA=X $ YT(II)=YA=Y                                   PEN0288
```

```
      CALL ELEV $ ZT(II)=ZA                                             PEN0289
      GO TO 230                                                         PEN0290
  520 IM=II $ ZH(1)=Z                                                   PEN0291
      PUNCH 3030                                                        PEN0292
 3030 FORMAT(7HCONTOUR)                                                 PEN0293
  530 CALL CONTOUR                                                      PEN0294
  540 DO 550 I=1,IM                                                     PEN0295
      X=XT(I) $ Y=YT(I) $ Z=ZH(I)                                       PEN0296
      IF(IPUNCH)545,550,545                                             PEN0297
  545 PUNCH 4000,X,Y,Z,ZT(I)                                            PEN0298
  550 CALL RANGE                                                        PEN0299
C                                                                       PEN0300
C     COMPUTE THE SURVIVAL PROBABILITY OF THE A/C.                      PEN0301
  600 TPS=1.                                                            PEN0302
      DO 802 I=1,IMAX $ JMAXX=JMAX(I)                                   PEN0303
      DO 802 J=1,JMAXX                                                  PEN0304
  802 TPS=TPS*PS(I,J)                                                   PEN0305
      PRINT 2050                                                        PEN0306
 2050 FORMAT(/////     8X,65HSUMMARY OF TIME IN VIEW AND SURVIVAL PROBABPEN0307
     *ILITY BY WEAPON NUMBER  )                                         PEN0308
      PRINT 2060                                                        PEN0309
 2060 FORMAT(1H0,13HWEAPON NUMBER)                                      PEN0310
  778 DO 780 I=1,IMAX                                                   PEN0311
      JMAXX=JMAX(I)                                                     PEN0312
  780 PRINT 2070,(NW(I,J),J=1,JMAXX)                                    PEN0313
 2070 FORMAT(21X,10I8)                                                  PEN0314
      PRINT 2080                                                        PEN0315
 2080 FORMAT(1H0,12HTIME IN VIEW)                                       PEN0316
      DO 781 I=1,IMAX                                                   PEN0317
      JMAXX=JMAX(I)                                                     PEN0318
  781 PRINT 2070,(ITIV(I,J),J=1,JMAXX)                                  PEN0319
      PRINT 2090                                                        PEN0320
 2090 FORMAT(1H0,20HSURVIVAL PROBABILITY)                               PEN0321
      DO 801 I=1,IMAX $ JMAXX=JMAX(I)                                   PEN0322
  801 PRINT 3000,(PS(I,J),J=1,JMAXX)                                    PEN0323
 3000 FORMAT(21X,10F8.4)                                                PEN0324
```

```
      PRINT 3010,TPS
 3010 FORMAT(///30X,26HTOTAL SURVIVAL PROBABILITY ,F6.4/1H1)
      END FILE 52
      WRITE (1,3025)
 3025 FORMAT(19H PRESS START SWITCH )
      PAUSE 1
 9990 CONTINUE
 9992 STOP
      END



      SUBROUTINE RANGE
C
C     THIS SUBROUTINE DETERMINES THE SURVIVAL PROBABILITY OF THE A/C.
C
      COMMON/RANG1/XW(10,20),YW(10,20),ZW(10,20),IMAX,JMAX(10),
     A           TIV(10,20),RMAX(10),TA(10),RLT(10),FT(10),RMIN(10),
     B           X,Y,Z,XWA,XWB,YWA,YWB
     1     /RANG2/ITIV(10,20),PS(10,20),JMARK,
     A           P(10,20),FTA(10)
     2     /ELEV1/PHI(12,25,25),MMAX(12),NMAX(12),BCX1,BCX2,BCX3,BCX4,
     A           BCY1,BCY2,BCY3,XL,YL,ZT,BCX,BCY
      TYPE INTEGER TIV,TA,RLT,FTA
      DO 10 I=1,IMAX $ JMAXX=JMAX(I)
      DO 10 J=1,JMAXX
      ICJ=I+10*(J-1)
      XA=(X-XW(ICJ))**2 $ YA=(Y-YW(ICJ))**2 $ ZA=(Z-ZW(ICJ))**2
C
C     DETERMINES IF THE TERRAIN INTERCEPTS THE LINE OF SIGHT.
       RW=10.   $ TR=SQRTF(XA+YA) $ TAN=(Z-ZW(ICJ))/TR
      SIN=(XW(ICJ)-X)/TR $ COS=(YW(ICJ)-Y)/TR
   20 ZL=RW*TAN+ZW(ICJ) $ XL=XW(ICJ)-RW*SIN $ YL=YW(ICJ)-RW*COS
      CALL ELEV
      IF(ZT.GT.ZL)70,40
```

```
C
   70 TIV(ICJ)=0 $ GO TO 10
   40 RW=RW+50.
      IF(RW.GT.TR)50,20
   50 TIV(ICJ)=TIV(ICJ)+1
      ITIV(ICJ)=ITIV(ICJ)+1
      IF(TIV(ICJ).LE.TA(I))10,80
   80 RP=XA+YA+ZA
      IF(RP.GT.RMAX(I))83,86
   83 TIV(ICJ)=TIV(ICJ)-1 $ GO TO 10
   86 IF(RP.GT.RMIN(I))81,83
   81 IF(TIV(ICJ).GT.FTA(I))82,84
   82 TIV(ICJ)=TA(I)/2-RLT(I) $ GO TO 10
C
C     THE APPROPIATE SURVIVAL PROBABILITY IS SELECTED.
C
   84 K=SQRTF(RP)/101
   99 PS(ICJ)=PS(ICJ)*P(I,K)
      JMARK=2 $ XWB=XW(ICJ) $ YWB=YW(ICJ)
   10 CONTINUE
   60 RETURN
      END




      SUBROUTINE ELEV
C
C     THIS SUBROUTINE DETERMINES THE TERRAIN ELEVATION, GIVEN X AND Y.
C
      COMMON/ELEV1/PHI(12,25,25),MMAX(12),NMAX(12),BCX1,BCX2,BCX3,BCX4,
     A               BCY1,BCY2,BCY3,XT,YT,ZT,BCX,BCY
C
C     DETERMINE WHICH POLYNOMIAL TO SOLVE.
C
      INX=(XT-BCX)/1000. $ INY=(YT-BCY)/1000.
```

```
      I=INX+4*(INY-1)                                                   PEN0397
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12)I                               PEN0398
    1 X=XT-BCX1        $ Y=YT-BCY1        $ GO TO 30                    PEN0399
    2 X=XT-BCX2        $ Y=YT-BCY1        $ GO TO 30                    PEN0400
    3 X=XT-BCX3        $ Y=YT-BCY1        $ GO TO 30                    PEN0401
    4 X=XT-BCX4        $ Y=YT-BCY1        $ GO TO 30                    PEN0402
    5 X=XT-BCX1        $ Y=YT-BCY2        $ GO TO 30                    PEN0403
    6 X=XT-BCX2        $ Y=YT-BCY2        $ GO TO 30                    PEN0404
    7 X=XT-BCX3        $ Y=YT-BCY2        $ GO TO 30                    PEN0405
    8 X=XT-BCX4        $ Y=YT-BCY2        $ GO TO 30                    PEN0406
    9 X=XT-BCX1        $ Y=YT-BCY3        $ GO TO 30                    PEN0407
   10 X=XT-BCX2        $ Y=YT-BCY3        $ GO TO 30                    PEN0408
   11 X=XT-BCX3        $ Y=YT-BCY3        $ GO TO 30                    PEN0409
   12 X=XT-BCX4        $ Y=YT-BCY3                                      PEN0410
C                                                                       PEN0411
C     SOLVE THE POLYNOMIAL, ZCOMP =.F(X,Y)                              PEN0412
C                                                                       PEN0413
   30 ZCOMP=0.                                                          PEN0414
      ICJ=I+12*(MMAX(I)-1)                                              PEN0415
   40 ICJCK=ICJ+300*(NMAX(I)-1) $ POLY=PHI(ICJCK)                       PEN0416
   50 ICJCK=ICJCK-300 $ POLY=POLY*Y+PHI(ICJCK)                          PEN0417
      IF(ICJCK-ICJ)100,70,50                                            PEN0418
   70 ZCOMP=ZCOMP*X+POLY $ IF(ICJ-I)100,90,80                           PEN0419
   80 ICJ=ICJ-12 $ GO TO 40                                             PEN0420
C                                                                       PEN0421
C     CORRECT FEET TO METERS                                            PEN0422
C                                                                       PEN0423
   90 ZT=0.304801*ZCOMP $ RETURN                                        PEN0424
  100 PRINT 1000                                                        PEN0425
 1000 FORMAT(1H1,5X,31HERROR IN ELEVATION COMPUTATION.  )               PEN0426
      STOP                                                              PEN0427
      END                                                               PEN0428
                                                                        PEN0429
                                                                        PEN0430
                                                                        PEN0431
                                                                        PEN0432
```

```
      SUBROUTINE TURN                                                   PEN0433
C                                                                       PEN0434
C     THIS SUBROUTINE TURNS THE AIRCRAFT AWAY FROM A FIRING WEAPON.     PEN0435
C                                                                       PEN0436
      COMMON/RANG1/XW(10,20),YW(10,20),ZW(10,20),IMAX,JMAX(10),         PEN0437
     A            TIV(10,20),RMAX(10),TA(10),RLT(10),FT(10),RMIN(10),   PEN0438
     B            X,Y,Z,XWA,XWB,YWA,YWB                                 PEN0439
     3      /TURN1/DELTH,R,DELX,DELY,DELD,PSI,IFLAG,KFLAG,LFLAG,MFLAG,  PEN0440
     A            COSDT,SINDT                                           PEN0441
      GO TO (10,30,300)IFLAG                                            PEN0442
   10 IF(PSI.GT.1.57079632)20,40                                        PEN0443
   20 PSI=PSI-1.57079632 $ IFLAG=2                                      PEN0444
   30 X1=-R*SINF(PSI) $ Y1=R*COSF(PSI) $ GO TO 50                       PEN0445
   40 X1=R*COSF(PSI) $ Y1=R*SINF(PSI)                                   PEN0446
   50 X2=X1*COSDT-Y1*SINDT $ Y2=Y1*COSDT+X1*SINDT                       PEN0447
   70 GO TO (80,90) MFLAG                                               PEN0448
   80 DELY=Y2-Y1 $ GO TO (100,110)LFLAG                                 PEN0449
   90 DELY=Y1-Y2 $ GO TO (100,110) LFLAG                                PEN0450
  100 DELX=X1-X2 $ GO TO 130                                            PEN0451
  110 DELX=X2-X1                                                        PEN0452
  130 IF(XWA.EQ.X) 140,150                                              PEN0453
  140 B=0.78539816 $ GO TO (190,200,230,240) KFLAG                      PEN0454
  150 B=ATANF((YWA-Y)/(XWA-X))                                          PEN0455
  160 IF(XWA.GT.X) 170,180                                              PEN0456
  170 B=B+3.14159264 $ GO TO (190,200,230,240) KFLAG                    PEN0457
  180 B=B+6.28318528 $ GO TO (190,200,230,240) KFLAG                    PEN0458
  190 KFLAG=2 $ GO TO 250                                               PEN0459
  200 IF(B.GT.A) 210,220                                                PEN0460
  210 KFLAG=3 $ GO TO 250                                               PEN0461
  220 KFLAG=4 $ GO TO 250                                               PEN0462
  230 IF(B.GT.A) 250,260                                                PEN0463
  240 IF(B.LT.A) 250,260                                                PEN0464
  250 A=B $ PSI=PSI+DELTH $ RETURN                                      PEN0465
  260 IFLAG=3                                                           PEN0466
  300 RETURN                                                            PEN0467
      END                                                               PEN0468
```

```
      SUBROUTINE CONTOUR
C
C     THIS SUBROUTINE COMPUTES A NAP OF THE EARTH FLIGHT PATH.
C
      COMMON/CONT1/AC,AD,IM,XT(300),YT(300),ZT(300),Z(300)
      IMARK=1
      DO 7 I=2,IM
      IF(IM.LE.I)10,9
    9 GO TO (10,11)IMARK
   10 IMARK=2
      GO TO 19
   11 IF(ZT(I).LE.ZT(I-1))19,17
   17 F=ZT(I)+3.
      IF(ZT(I).LE.ZT(I+1))20,18
   20 IMARK=2 $ GO TO 21
   18 IMARK=1 $ GO TO 21
   19 F=ZT(I)+10.
   21 S=(F-Z(I-1))/SQRTF((XT(I)-XT(I-1))**2+(YT(I)-YT(I-1))**2)
      IF(AC.GE.S)15,22
   15 IF(S.GE.AD) 100,22
  100 Z(I)=F $ GO TO 7
   22 IF(S)43,44,44
   43 Z(I)=Z(I-1)+AD*SQRTF((XT(I)-XT(I-1))**2+(YT(I)-YT(I-1))**2)
      GO TO 7
   44 Z(I)=F
      J=0
   46 J=J+1
      Z(I-J)=F-AC*SQRTF((XT(I-J)-XT(I-J-1))**2+(YT(I-J)-YT(I-J-1))**2)
      IF(I-J-1)48,48,47
   47 IF(((Z(I-J)-Z(I-J-1))/SQRTF((XT(I-J)-XT(I-J-1))**2+(YT(I-J)-
     *YT(I-J-1))**2)).GT.AC)45,7
   45 F=Z(I-J) $ GO TO 46
```

```
   48 PRINT 7010                                                              PEN0505
 7010 FORMAT(1H1////78H AIRCRAFT BEGINNING ELEVATION IS TO LOW TO AVOID PEN0506
     *TERRAIN. THIS RUN TERMINATED.  )                                        PEN0507
      STOP                                                                    PEN0508
    7 CONTINUE                                                                PEN0509
      RETURN                                                                  PEN0510
      END                                                                     PEN0511
         END                                                                  PEN0512
         FINIS                                                                PEN0513
```

Appendix III

Inputs and Outputs of PENAIR

Data cards for the PENAIR program should be prepared in the following manner

Data Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
1	1	NRUN	1-10	A right justified integer, the number of runs to be performed.
2	1	KMAX	1-10	A right justified integer, the number of squares used to approximate the terrain, KMAX ≤ 12.
		XMIN	11-20	A right justified floating point number, the x coordinate of the left edge of the terrain. All coordinates are in meters, that is the distance between the two x coordinates 72455 and 72454 is one meter.
		YMIN	21-30	A right justified floating point number, the y coordinate of the bottom edge of the terrain.
		XMAX	31-40	The x coordinate of the right edge of the terrain.

[illegible]

Date Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
		YMAX	41-50	The y coordinate of the top edge of the terrain.
		TI	51-60	A right justified floating point number, the time interval of the game (seconds), TI > 0.
		TPD	61-70	A right justified floating point number, the dimension of the map square approximated by one polynomial (meters).
3	1	X Y Z	1-10 11-20 21-30	Right justified floating point numbers, the beginning coordinates of the aircraft. If IMARK = 3, 4, or 6, y must equal YMIN.
		X2 Y2	31-40 41-50	The x and y coordinates of the desired destination of the aircraft. If IMARK = 7 or 8, this card may be blank.
4	1	GMAX	1-10	A right justified floating point number, the maximum positive G force to be used by the aircraft.
		RC	11-20	A right justified floating point number, the aircraft's maximum rate of climb (feet/minute)

Data Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
		RD	21-30	A right justified floating point number, the aircraft's maximum rate of descent (feet/minute).
		GS	31-40	A right justified floating point number, the original airspeed of the aircraft.
5	1	IMAX	1-10	A right justified integer, the number of types of weapons, IMAX $\leq$ 10.
6	1	WPNAWPNB	1-16	The name of the weapon, e.g., 50 Cal. MG
		JMAX	21-25	A right justified integer, the number of weapons of this type, JMAX $\leq$ 20.
		TA	26-30	A right justified integer, the time required to acquire the target by this weapon (seconds), TA $\geq$ 0.
		FT	31-35	A right justified integer, the firing time of this weapon before reloading is required (seconds), FT > 0.

Data Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
		RLT	36-40	A right justified integer, the time to reload this weapon (seconds) RLT ≥ 0.
		RMAX	41-50	A right justified floating point number, the maximum range of this weapon(meters) RMAX > 0 .
		RMIN	51-60	A right justified floating point number, the minimum range of this weapon(meters). RMIN ≥ 0 .
7	2	P	1-8 9-16 72-80	The probability of kill by this weapon in one time interval at ranges of 100, 200, 300, ..., 1000 meters.
			1-8 9-16 72-80	The second card of this group contains the kill probabilities at ranges of 1100 to 2000 meters in 100 meter intervals.
8	JMAX	NW	1-2	The number assigned to the weapon, $1 \leq NW \leq 99$.
		XW YW ZW	11-20 21-30 31-40	The x, y and z coordinates of this weapon, ZW may be blank since PENAIR will compute the weapon's elevation.

NOTE: Data groups 6, 7 and 8 should be repeated IMAX times.

Data Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
9	1	MMAX	1-10	A right justified integer, the degree of x in the polynomial that approximates the first map square.
		NMAX	11-20	A right justified integer, the degree of y in the polynomial that approximates the first map square
10	As many cards as necessary	PHI	1-80	The coefficients of the first polynomial appear on this and the following cards of this data group in a 4E20.10 field as obtained from the terrain simulation program.

NOTE: Data groups 9 and 10 should be repeated KMAX times.

11	4	AFIELD	1-16	These four cards contain the four lines of the title to be printed at the top of the output.

[illegible]

Data Group	Number of Cards	Variable Name	Columns of Cards	Description of Variable
12	1	IMARK	10	The·value of this flag, determines the flightpath to be flown. IMARK = 1, upon receiving fire, fly straight and level, = 2, climb straight ahead, = 3, turn away, level, = 4, turn away and climb, = 5, dive and fly nap of the earth, = 6, turn away and fly nap of the earth, = 7, the flight path is completely predetermined and read in as data group 13, = 8, the x and y coordinates are read in as data group 13, and a nap of the earth flightpath is computed.
		IPUNCH	20	IPUNCH = 1, the flightpath will will be punched on cards. = 0, the flightpath will not be punched.

Data Group	Number of Cards	Variable Name	Column of Card	Description of Variable
		GSNEW	21-30	A right justified floating point number, the ground speed to be attained by the aircraft after receiving fire.

NOTE: Data groups 11 and 12 will be repeated NRUN times.

Data Group	Number of Cards	Variable Name	Column of Card	Description of Variable
13	As many cards as necessary but not greater than 300.	NPATH	6-7	Any positive integer, used to identify the flightpath,
		XT	18-26	Right justified floating point numbers, each of the cards of this group contain x, y, and z coordinates of the aircraft in consecutive time intervals.
		YT	27-35	
		ZH	36-44	

NOTE: A blank card should be inserted at the end of this data group. This data group will be used only if IMARK = 7 or 8. If IMARK = 8, ZH may be blank. Data group 13 should follow its corresponding data groups 11 and 12.

Output of the PENAIR Program

A sample output from the PENAIR program is illustrated in Figures 14 and 15. The input parameters of the weapons are printed with the weapon positions and elevations. The second page of the output (Figure 15) is a summary of time in view and aircraft survival probability. The time in view listed here is in number of time intervals in view. If the time interval used in this example had been 0.5 second, the aircraft was in view of weapon number 49, located at (70080, 76940, 339), for 28 seconds (56 x 0.5). The survival probability of the aircraft from this weapon was .9991.

SUMMARY OF WEAPON DATA

WEAPON TYPE	M14 SQUAD	.50 CAL. MG	M-42	QUAD 50
NUMBER OF WEAPONS	4	4	3	2
ACQUISITION TIME (SEC)	6	6	6	6
FIRING TIME (SEC)	10	15	9000	15
RELOADING TIME (SEC)	6	8	0	8
MINIMUM RANGE (METERS)	0	100	100	100
MAXIMUM RANGE (METERS)	600	1400	2000	2000

WEAPON TYPE	SITE NUMBER	WEAPON POSITIONS		
M14 SQUAD				
	1	72810	76180	300
	5	71880	76220	315
	9	71350	76620	359
	12	71020	76450	355
.50 CAL. MG				
	37	72420	75180	331
	38	70240	76050	357
	39	71020	76160	357
	41	72460	74880	243
M-42				
	42	71660	74640	311
	43	70070	75000	288
	49	70080	76940	339
QUAD 50				
	51	70840	75900	349
	52	71620	76000	375

FIGURE 14

FIRST PAGE OF OUTPUT FROM THE PENAIR PROGRAM

```
               SUMMARY OF
             RUN NUMBER 2-1-C
MISSION -    11 APRIL 1965
             FLY STRAIGHT AND LEVEL
             INCREASE SPEED TO 110 KNOTS


SUMMARY OF TIME IN VIEW AND SURVIVAL PROBABILITY BY WEAPON NUMBER

WEAPON NUMBER
                         1         5         9        12
                        37        38        39        41
                        42        43        49
                        51        52

TIME IN VIEW
                       100        85        38        14
                       100       100       100        47
                        88        37        56
                       100       100

SURVIVAL PROBABILITY
                    1.0000     .6696     .8653    1.0000
                     .9611     .9983     .9038     .9869
                     .0678     .8168     .9991
                     .7027     .1722

                  TOTAL SURVIVAL PROBABILITY .0033
```

FIGURE 15

SECOND PAGE OF OUTPUT FROM THE PENAIR PROGRAM

APPENDIX IV

FORTRAN Listing of the TERRAIN Program

TERRAIN consists of a main program and three overlays. The control cards for the COOP-Monitor of the CDC 1604 are included in this listing and are explained in Appendix V. The variable names used in the TERRAIN program are defined in the comments appearing in the FORTRAN listing. The variable names of input and output parameters are explained in Appendix V.

```
-COOP,,HARRISON,0/49/S/56/57/7/E/45=54,45,20000,,PROGRAM TERRAIN.      TER0001
-BINARY,56.                                                            TER0002
(MAIN,7.                                                               TER0003
-FTN,L,E.                                                              TER0004
      PROGRAM TERRAIN                                                  TER0005
      COMMON/M1/XFIRST,XLAST,YFIRST,YLAST,Z,IMAX,JMAX                  TER0006
     1/M2/MMAX,NMAX,NEWCTL                                             TER0007
     2      /PLOT1/ZTMIN,CND,NC,LABL(50),ITITLE(12)                    TER0008
     3      /PLOT2/ZCOMP(45,45),A,KK                                   TER0009
      DIMENSION Z(45,45)                                               TER0010
      CALL OVERLAY (7,0,1)                                             TER0011
      CALL OVERLAY (7,0,2)                                             TER0012
C
      IF(NEWCTL)10,20,10                                               TER0013
C     NEWCTL=0, NO MAP WILL BE DRAWN.
C           =1, A CONTOUR MAP USING THE APPROXIMATING POLYNOMIAL
C               WILL BE DRAWN.
C
   10 CALL OVERLAY(7,0,3)                                              TER0014
   20 STOP                                                             TER0015
      END                                                              TER0016
         FINIS                                                         TER0017
-BINARY,56.                                                            TER0018
(OVERLAY,7,1.                                                          TER0019
-FTN,L,E.                                                              TER0020
      PROGRAM MAPPING                                                  TER0021
C  FINDS THE ELEVATION OF EQUALLY SPACED GRID POINTS.                  TER0022
C  S=SPACING OF GRID LINES  A=LINEAR DIMENSION OF MAP RECTANGLE        TER0023
C  GX=X-COORDINATE OF SW CORNER OF MAP RECTANGLE                       TER0024
C  GY=Y-COORDINATE OF SW CORNER OF MAP RECTANGLE                       TER0025
C  XP(I),YP(I)=MEASURED COORDINATES OF RECTANGLE CORNERS.              TER0026
C  FZ=HEIGHT OF CONTOUR LINE  NC=CARD NO  X(I),Y(I)=COORDINATES OF     TER0027
C     POINTS ON A CONTOUR LINE.                                        TER0028
      COMMON/1/S,J,ZC/2/KM/M1/BCX,XLAST,BCY,YLAST,Z,IMAX,JMAX          TER0029
     1      /PLOT1/ZTMIN,CND,NCI,LABL(50),ITITLE(12)                   TER0030
     2      /PLOT2/ZCOMP(45,45),A ,KK                                  TER0031
```

```
    3      /M2/MMAX,NMAX,NEWCTL                                         TER0032
     DIMENSION XC(500),YC(500),YX(45, 70),ZX(45, 70),X(7),Y(7),XP(4),  TER0033
    1YP(4),XY(45, 70),ZY(45, 70),LX(70),LY(70),YXI(45,70),Z(45,45),     TER0034
    1XYI(45,70),ZXI(45,70),ZPX(45,45),ZPY(45,45),ZYI(45,70)             TER0035
     EQUIVALENCE(XC,YXI),(YC,ZXI),(YX,XYI),(ZX,ZYI),(XY,ZPX),(ZY,ZPY)  TER0036
    1,(LX,Z)                                                            TER0037
     CORRECT(X,Y,A,X1,X2,X3,Y1) = A*(X-(X-X1)*(Y-Y1)*(X3-X2)/((X3-X1)* TER0038
    1(Y3-Y1))-X1)/(X2-X1)                                               TER0039
     DO 300 L=1,50
 300 LABL(L)=4H.
     J=0                                                                TER0040
     READ 500,ITITLE                                                    TER0041
 500 FORMAT(6A8)                                                        TER0042
     READ 700,NL                                                        TER0043
 700 FORMAT(I5)                                                         TER0044
     DO 800 L=1,NL                                                      TER0045
 800 READ 600,II,(LABL(II))                                             TER0046
 600 FORMAT(I5,1X,A4)                                                   TER0047
     READ 900,NMAX,MMAX,NEWCTL                                          TER0048
 900 FORMAT(3I5)                                                        TER0049
   1 READ 100,S,A                                                       TER0050
 100 FORMAT(2F10)                                                       TER0057
     READ 200,ZTMIN,ZTMAX,CND                                           TER0051
 200 FORMAT(3F10.0)                                                     TER0052
     NCI=(ZTMAX-ZTMIN)/CND+1.                                           TER0053
     KM=(A+200.)/S+1.                                                   TER0054
     DO 30 K=1,KM  $  LX(K)=1                                           TER0055
  30 LY(K)=1                                                            TER0056
   2 READ 101,GX,GY,(XP(I),YP(I),I=1,4)                                 TER0058
 101 FORMAT(-3PF2,F2,6X,0P,8F5)                                         TER0059
     BCX=GX-100.  $  BCY=GY-100.                                        TER0060
   4 READ 102,FZ,NC,(X(I),Y(I),I=1,7)                                   TER0061
 102 FORMAT(4X,-1PF4,0P,I2,14F5)                                        TER0062
   5 IF(NC)14,14,6                                                      TER0063
   6 GO TO(7,8),NC                                                      TER0064
   7 ZC=FZ                                                              TER0065
```

```
  8 DO 13 I=1,7                                                          TER0066
 10 IF(X(I)+Y(I))12,11,12                                                TER0067
 12 XC(I+J)=X(I)                                                         TER0068
 13 YC(I+J)=Y(I)                                                         TER0069
    J=J+7                                                                TER0070
    GO TO 4                                                              TER0071
 11 J=J+I-1                                                              TER0072
    GO TO 4                                                              TER0073
 14 X1=XP(1)  $  Y1=YP(1)  $  X2=XP(2)  $  Y2=YP(2)  $  X3=XP(3)         TER0074
    Y3=YP(3)  $  Y4=YP(4)                                                TER0075
    DO 17 I=1,J                                                          TER0076
    XCI=XC(I)  $  YCI=YC(I)                                              TER0077
 16 XC(I)=CORRECT(XCI,YCI,A,X1,X2,X3,Y1   )+GX                           TER0078
    YC(I)=CORRECT(YCI,XCI,A,Y1,Y4,Y3,X1   )+GY                           TER0079
 17 CONTINUE                                                             TER0080
 19 CALL CONTOURI(XC,YC,BCX,LX,YX,ZX)                                    TER0081
 20 CALL CONTOURI(YC,XC,BCY,LY,XY,ZY)                                    TER0082
    IF(FZ.LT.9990.)3,31                                                  TER0083
  3 READ 102,FZ,NC,(X(I),Y(I),I=1,7)                                     TER0084
    J=0                                                                  TER0085
    IF(NC)2,2,7                                                          TER0086
 31 CALL ORDER(YX,ZX,LX,YXI,ZXI)                                         TER0087
 34 CALL ORDER(XY,ZY,LY,XYI,ZYI)                                         TER0088
 37 CALL XPOLATE(YXI,ZXI,BCY,LX,ZPX)                                     TER0089
 38 CALL XPOLATE(XYI,ZYI,BCX,LY,ZPY)                                     TER0090
 39 DO 40 K=1,KM                                                         TER0091
    DO 40 J=1,KM                                                         TER0092
    Z(K,J)=(ZPX(K,J)+ZPY(J,K))/2.                                        TER0093
    IF (Z(K,J).LT.ZTMIN)767,768                                          TER0094
767 Z(K,J)=ZTMIN $ GO TO 40                                              TER0095
768 IF (Z(K,J).GT.ZTMAX)769,40                                           TER0096
769 Z(K,J)=ZTMAX                                                         TER0097
 40 CONTINUE                                                             TER0098
    YLAST=BCY+A+200.                                                     TER0099
    XLAST=BCX+A+200.                                                     TER0100
    IMAX=KM  $  JMAX=KM                                                  TER0101
```

```
      END
      SUBROUTINE CONTOURI(XC,YC,BC,L,YX,ZX)
C     FINDS THE COORDINATES OF THE TERRAIN AT THE INTERSECTIONS OF
C     CONTOURS AND GRID LINES.
      COMMON/1/S,J,ZC
      DIMENSION XC(500),YC(500),L( 70),YX(45, 70),ZX(45, 70)
    1 FK=(XC(1)-BC)/S+1.  $  KF=FK
    2 IF(FK-KF)6,4,6
    4 YX(KF,1)=YC(1)  $  ZX(KF,1)=ZC  $  L(KF)=2
    6 DO 11 I=2,J
    7 K=(XC(I)-BC)/S+1.
    8 IF(K-KF)12,11,10
   12 K=K+1  $  KF=K-1  $  GO TO 9
   10 KF=K
    9 XXK=BC     +(K-1)*S
    . YX(K,L(K))=(YC(I)-YC(I-1))*(XXK-XC(I-1))/(XC(I)-XC(I-1))+YC(I-1)
      ZX(K,L(K))=ZC  $  L(K)=L(K)+1
   11 CONTINUE
      RETURN
         END
      SUBROUTINE ORDER(YX,ZX,L,YXI,ZXI)
C   ORDERS THE PAIRS OF VALUES YX(I,J),ZX(I,J) AFTER INCREASING YX(I,J) F
C   I
      COMMON/2/KM
      DIMENSION YX(45, 70),ZX(45, 70),L( 70),YXI(45, 70),ZXI(45, 70)
      KMI=KM-1
    1 DO 10 K=2,KMI $ LK=L(K)-1
    2 DO 10 I=1,LK
    3 FMIN=YX(K,1)  $  MF=1
    4 DO 7 M=2,LK
    5 IF(YX(K,M)-FMIN)6,6,7
    6 FMIN=YX(K,M)  $  MF=M
    7 CONTINUE
    8 YXI(K,I)=FMIN
    9 ZXI(K,I)=ZX(K,MF)
   10 YX(K,MF)=9999999.
```

```
      LK=L(1)-1                                                         TER0138
      DO 20 I=1,LK                                                      TER0139
      YXI(1,I)=YXI(2,I)                                                 TER0140
   20 ZXI(1,I)=ZXI(2,I)                                                 TER0141
      LK=L(KM-1)-1                                                      TER0142
      DO 30 I=1,LK                                                      TER0143
      YXI(KM,I)=YXI(KMI,I)                                              TER0144
   30 ZXI(KM,I)=ZXI(KMI,I)                                              TER0145
      RETURN                                                            TER0146
      END                                                               TER0147
      SUBROUTINE XPOLATE(YXI,ZXI,BCY,LX,ZPX)                            TER0148
C  FINDS HEIGHT OF THE TERRAIN AT EQUIDISTANT POINTS ALONG GRID LINES   TER0149
      COMMON/1/S,AA,BB/2/KM                                             TER0151
      DIMENSION YXI(45,70),ZXI(45,70),LX(70),ZPX(45,45)                 TER0152
    1 POL(X,XL,XH,ZL,ZH)=(X-XL)*(ZH-ZL)/(XH-XL)+ZL                      TER0153
   10 DO 190 K=1,KM                                                     TER0154
   20 LXK=LX(K)-1  $  KF=K  $  M=2                                      TER0155
   30 IF(LXK-1)60,40,120                                                TER0156
   40 DO 50 J=1,KM                                                      TER0157
   50 ZPX(K,J)=ZXI(K,1)  $  GO TO 190                                   TER0158
   60 IF(KF-KM/2)80,80,70                                               TER0159
   70 KF=KF-1  $  GO TO 90                                              TER0160
   80 KF =KF+1                                                          TER0161
   90 LXK=LX(KF)-1                                                      TER0162
  100 DO 110 J=1,LXK $  ZXI(K,J)=ZXI(KF,J)                              TER0163
  110 YXI(K,J)=YXI(KF,J)  $  GO TO 30                                   TER0164
  120 DO 170 J=1,KM                                                     TER0165
  130 YPX=BCY+(J-1)*S                                                   TER0166
  140 DO 160 I=M,LXK                                                    TER0167
  150 IF(YXI(K,I)-YPX)160,170,170                                       TER0168
  160 CONTINUE                                                          TER0169
  180 I=LXK  $  M=I+1                                                   TER0170
  170 ZPX(K,J)=POL(YPX,YXI(K,I-1),YXI(K,I),ZXI(K,I-1),ZXI(K,I))         TER0171
  190 CONTINUE                                                          TER0172
      RETURN                                                            TER0173
      END                                                               TER0174
```

```
      FINIS                                                            TER0175
-BINARY,56.                                                            TER0176
(OVERLAY,7,2.                                                          TER0177
-FTN,L,E.                                                              TER0178
      PROGRAM SURFIT                                                   TER0179
C     FITS A POLYNOMIAL TO THE POINTS COMPUTED IN OVERLAY 1.
      DIMENSION DENPA(45),DENQA(45),ALPHA(45,45),P(45,45),Q(45,45),    TER0180
     1PC(45,45),QC(45,45),A(45),B(45),C(45),D(45),X(45),Y(45),         TER0181
     2Z(45,45),BETA(45,45),PHI(45,45),             ZDIFF(45,45)        TER0182
      COMMON/M1/XFIRST,XLAST,YFIRST,YLAST,Z,IMAX,JMAX                  TER0183
     2      /PLOT1/ZTMIN,CND,NC,LABL(50),ITITLE(12)                    TER0184
     1      /PLOT2/ZCOMP(45,45),AQ,KK                                  TER0185
     1/M2/MMAX,NMAX,NEWCTL                                             TER0186
      EQUIVALENCE(P,PC),(Q,QC),(ALPHA,ZCOMP)                           TER0187
      U=1.                                                             TER0188
      W=1.                                                             TER0189
      XNUM=IMAX-1                                                      TER0190
      DELTAX=(XLAST-XFIRST)/XNUM                                       TER0191
      II=IMAX-1                                                        TER0192
      X(1)=XFIRST                                                      TER0193
      DO 62 I=2,II                                                     TER0194
   62 X(I)=X(I-1)+DELTAX                                               TER0195
      X(IMAX)=XLAST                                                    TER0196
      YNUM=JMAX-1                                                      TER0197
      DELTAY=(YLAST-YFIRST)/YNUM                                       TER0198
      JJ=JMAX-1                                                        TER0199
      Y(1)=YFIRST                                                      TER0200
      DO 72 J=2,JJ                                                     TER0201
   72 Y(J)=Y(J-1)+DELTAY                                               TER0202
      Y(JMAX)=YLAST                                                    TER0203
      PRINT 2113,ITITLE                                                TER0204
 2113 FORMAT(1H1,//////(20X,6A8)////)                                  TER0205
C     PRINT INPUT DATA X,Y,AND Z                                       TER0206
 3010 PRINT 2100                                                       TER0207
 2100 FORMAT(////10X,44HINPUT DATA GRID COORDINATES X DIRECTION X(I)//) TER0208
      PRINT2101,( X(I), I=1,IMAX )                                     TER0209
```

```
 2101 FORMAT(4F14.0)                                                        TER0210
      PRINT 2102                                                            TER0211
 2102 FORMAT(/// 10X,44HINPUT DATA GRID COORDINATES Y DIRECTION Y(J)//)     TER0212
      PRINT2101,(Y(J),J=1,JMAX)                                             TER0213
      PRINT 2103                                                            TER0214
 2103 FORMAT(1H1////20X,33HINPUT DATA GRID ELEVATIONS Z(I,J)///)            TER0215
      PRINT2105,(J, J=1,JMAX)                                               TER0216
 2105 FORMAT(4H I/J I13,3I14 / (4X,I13,3I14))                               TER0217
      DO2106 I=1,IMAX                                                       TER0218
 2106 PRINT2107,I,(Z(I,J),J=1,JMAX)                                         TER0219
 2107 FORMAT(// I3,4F14.0 /(3X4F14.0))                                      TER0220
C NORMALIZATION OF VARIABLES                                                TER0221
      SUMX=0.                                                               TER0222
      SUMY=0.                                                               TER0223
      SUMZ=0.                                                               TER0224
      DO 10 I=1,IMAX                                                        TER0225
   10 SUMX=SUMX+X(I)                                                        TER0226
      XIMAX=IMAX                                                            TER0227
      XMEANX= SUMX/XIMAX                                                    TER0228
      DO 20 I=1,IMAX                                                        TER0229
   20 X(I)=X(I)-XMEANX                                                      TER0230
      DO 30 J=1,JMAX                                                        TER0231
   30 SUMY=SUMY+Y(J)                                                        TER0232
      XJMAX=JMAX                                                            TER0233
      YMEANY=SUMY/XJMAX                                                     TER0234
      DO 40 J=1,JMAX                                                        TER0235
   40 Y(J)=Y(J)-YMEANY                                                      TER0236
      DO 50 I=1,IMAX                                                        TER0237
      DO 50 J=1,JMAX                                                        TER0238
   50 SUMZ=SUMZ+Z(I,J)                                                      TER0239
      XIJMAX=IMAX*JMAX                                                      TER0240
      ZMEANZ=SUMZ/XIJMAX                                                    TER0241
      DO 60 I=1,IMAX                                                        TER0242
      DO 60 J=1,JMAX                                                        TER0243
   60 Z(I,J)=Z(I,J)-ZMEANZ                                                  TER0244
C EVALUATE ORTHOGONAL POLYNOMIALS                                           TER0245
```

```
      XNUMA=0.                                                          TER0246
      DENA=0.                                                           TER0247
      DO 70 I=1,IMAX                                                    TER0248
      P(1,I)=1.                                                         TER0249
      XNUMA=XNUMA+     X(I)                                             TER0250
   70 DENA=DENA+U                                                       TER0251
      A(2)=XNUMA/DENA                                                   TER0252
      DO 80  I=1,IMAX                                                   TER0253
   80 P(2,I)=X(I)-A(2)                                                  TER0254
      DO 90 N=3,NMAX                                                    TER0255
      XNUMA=0.                                                          TER0256
      DENA=0.                                                           TER0257
      DENB=0.                                                           TER0258
      DO 85 I=1,IMAX                                                    TER0259
      XNUMA=XNUMA+     X(I)*P(N-1,I)**2                                 TER0260
      DENA=DENA+     P(N-1,I)**2                                        TER0261
   85 DENB=DENB+     P(N-2,I)**2                                        TER0262
      A(N)=XNUMA/DENA                                                   TER0263
      B(N)=DENA/DENB                                                    TER0264
      DO 90 I=1,IMAX                                                    TER0265
   90 P(N,I)=(X(I)-A(N))*P(N-1,I)-B(N)*P(N-2,I)                         TER0266
      XNUMC=0.                                                          TER0267
      DENC=0.                                                           TER0268
      DO 100 J=1,JMAX                                                   TER0269
      Q(1,J)=1.                                                         TER0270
      XNUMC=XNUMC+     Y(J)                                             TER0271
  100 DENC=DENC+W                                                       TER0272
      C(2)=XNUMC/DENC                                                   TER0273
      DO 110 J=1,JMAX                                                   TER0274
  110 Q(2,J)=Y(J)-C(2)                                                  TER0275
      DO 120 M=3,MMAX                                                   TER0276
      XNUMC=0.                                                          TER0277
      DENC=0.                                                           TER0278
      DEND=0.                                                           TER0279
      DO 115 J=1,JMAX                                                   TER0280
      XNUMC=XNUMC+     Y(J)*Q(M-1,J)**2                                 TER0281
```

```
      DENC=DENC+      Q(M-1,J)**2                                       TER0282
  115 DEND=DEND+      Q(M-2,J)**2                                       TER0283
      C(M)=XNUMC/DENC                                                   TER0284
      D(M)=DENC/DEND                                                    TER0285
      DO 120 J=1,JMAX                                                   TER0286
  120 Q(M,J)=(Y(J)-C(M))*Q(M-1,J)-D(M)*Q(M-2,J)                         TER0287
      DO 130 N=1,NMAX                                                   TER0288
      DENPA(N)=0.                                                       TER0289
      DO 130 I=1,IMAX                                                   TER0290
  130 DENPA(N)=DENPA(N)+      P(N,I)**2                                 TER0291
      DO 140 M=1,MMAX                                                   TER0292
      DENQA(M)=0.                                                       TER0293
      DO 140 J=1,JMAX                                                   TER0294
  140 DENQA(M)=DENQA(M)+      Q(M,J)**2                                 TER0295
      DO 160 N=1,NMAX                                                   TER0296
      DO 160 M=1,MMAX                                                   TER0297
      ALPH=0.                                                           TER0298
      DO 150 I=1,IMAX                                                   TER0299
      DO 150 J=1,JMAX                                                   TER0300
      ALPH=ALPH+           Z(I,J)*P(N,I)*Q(M,J)                         TER0301
  150 CONTINUE                                                          TER0302
      ALPHA(N,M)=ALPH/(DENPA(N)*DENQA(M))                               TER0303
  160 BETA(N,M)=ALPHA(N,M)*ALPH                                         TER0304
C APPLICATION OF GAUSS CRITERION TO DETERMINE THE DEGREE POLY WHICH     TER0305
C  YIELDS THE CLOSEST FIT TO THE GIVEN DATA                             TER0306
      SUMZSQ=0.                                                         TER0307
      DO 170 I=1,IMAX                                                   TER0308
      DO 170 J=1,JMAX                                                   TER0309
  170 SUMZSQ=SUMZSQ+           Z(I,J)**2                                TER0310
      IS=1                                                              TER0311
      IT=1                                                              TER0312
      DO 190 N=1,NMAX                                                   TER0313
      BETASUM=0.                                                        TER0314
      DO 190 M=1,MMAX                                                   TER0315
      XX=IMAX*JMAX-N*M                                                  TER0316
      DO 175 L=1,N                                                      TER0317
```

```
  175 BETASUM=BETASUM+BETA(L,M)                                      TER0318
      IF( BETASUM-SUMZSQ)   172,172,171                              TER0319
  171 TRIGAUS=0.                                                     TER0320
      GO TO 174                                                      TER0321
  172 IF (IMAX*JMAX-N*M) 2172,171,2172                               TER0322
 2172 TRIGAUS=(SUMZSQ-BETASUM)/XX                                    TER0323
  174 IF(N-1) 176,375,176                                            TER0324
  375 IF(M-1) 176,177,176                                            TER0325
  177 GAUSCRT=TRIGAUS                                                TER0326
      GO TO 190                                                      TER0327
  176 IF(GAUSCRT - TRIGAUS) 180,178,180                              TER0328
  178 IF(N*M-IS*IT) 179,180,180                                      TER0329
  179 IS=N                                                           TER0330
      IT=M                                                           TER0331
  180 CONTINUE                                                       TER0332
      IF( GAUSCRT - TRIGAUS) 190,190,181                             TER0333
  181 GAUSCRT=TRIGAUS                                                TER0334
      IS=N                                                           TER0335
      IT=M                                                           TER0336
  190 CONTINUE                                                       TER0337
      NMAX=IS                                                        TER0338
      MMAX=IT                                                        TER0339
      TEMP=(IMAX*JMAX-NMAX*MMAX)/(IMAX*JMAX)                         TER0340
      XMINSQD=SQRTF( GAUSCRT*TEMP)                                   TER0341
C EVALUATION OF ORTHO POLY COEFFICIENTS                              TER0342
      DO 200 N=1,NMAX                                                TER0343
      PC(N,N)=1.                                                     TER0344
      NM1=N-1                                                        TER0345
      DO 200 IS=1,NM1                                                TER0346
      PC(N,IS) = -A(N)* PC(NM1,IS)                                   TER0347
      IF(IS-1) 197,198,197                                           TER0348
  197 PC(N,IS)= PC(N,IS) +PC(N-1,IS-1)                               TER0349
  198 IF(IS-NM1)  199,200,199                                        TER0350
  199 PC(N,IS)=PC(N,IS) - B(N)* PC(N-2,IS)                           TER0351
  200 CONTINUE                                                       TER0352
      DO 210 M=1,MMAX                                                TER0353
```

```
      QC(M,M)=1.                                                        TER0354
      MM1=M-1                                                           TER0355
      DO 210 IT=1,MM1                                                   TER0356
      QC(M,IT)= -C(M)* QC(MM1,IT)                                       TER0357
      IF(IT-1)207,208,207                                               TER0358
  207 QC(M,IT)= QC(M,IT)+ QC(MM1,IT-1)                                  TER0359
  208 IF(IT-MM1) 209,210,209                                            TER0360
  209 QC(M,IT)=QC(M,IT)-D(M)*QC(M-2,IT)                                 TER0361
  210 CONTINUE                                                          TER0362
C EVALUATION OF APPROXIMATING POLY COEFF                                TER0363
      DO 230 IS=1,NMAX                                                  TER0364
      DO 220 IT=1,MMAX                                                  TER0365
      PHI(IS,IT)=0.                                                     TER0366
      DO 220 N=IS,NMAX                                                  TER0367
      DO 220 M=IT,MMAX                                                  TER0368
  220 PHI(IS,IT)=PHI(IS,IT)+ALPHA(N,M)*PC(N,IS)*QC(M,IT)                TER0369
  230 CONTINUE                                                          TER0370
C EVALUATE OF DEPENDENT VARIABLES USING THE APPROXIMATING POLY          TER0371
      CMINSQD=0.                                                        TER0372
      CSUMDIF=0.                                                        TER0373
      CMAXDIF=0.                                                        TER0374
      DO 250 I=1,IMAX                                                   TER0375
      DO 250 J=1,JMAX                                                   TER0376
      ZCOMP(I,J)=0.                                                     TER0377
  240 IS=NMAX                                                           TER0378
  248 POLY=PHI(IS,MMAX)                                                 TER0379
  241 IT=MMAX-1                                                         TER0380
  249 POLY=POLY*Y(J)+PHI(IS,IT)                                         TER0381
      IF(IT-1) 1999,247,242                                             TER0382
  242 IT=IT-1                                                           TER0383
      GO.TO 249                                                         TER0384
  247     ZCOMP(I,J)=ZCOMP(I,J)*X(I)+POLY                               TER0385
  243 IF(IS-1) 2000,245,244                                             TER0386
  244 IS=IS-1                                                           TER0387
      GO TO 248                                                         TER0388
  245 RESCOMP=Z(I,J)-ZCOMP(I,J)                                         TER0389
```

```
      Z(I,J)=Z(I,J)+ZMEANZ                                              TER0390
      ZCOMP(I,J)=ZCOMP(I,J)+ZMEANZ                                      TER0391
      CMINSQD=CMINSQD+           RESCOMP**2                             TER0392
      ABRESCP=ABSF(RESCOMP)                                             TER0393
      IF(ABRESCP-CMAXDIF) 250,250,246                                   TER0394
  246 CMAXDIF=ABRESCP                                                   TER0395
  250 CONTINUE                                                          TER0396
      PRINT 1251                                                        TER0397
 1251 FORMAT(1H1////47HCOMPUTED GRID ELEVATIONS Z(I,J) AS EVALUATED BY  TER0398
     1 /12X,24HAPPROXIMATING POLYNOMIAL ///)                            TER0399
      PRINT 2105,(J,J=1,JMAX)                                           TER0400
      DO 260 I=1,IMAX                                                   TER0401
  260 PRINT 2107, I, (ZCOMP(I,J),J=1,JMAX)                              TER0402
      CMINSQD=SQRTF(CMINSQD/XIJMAX    )                                 TER0403
      CSUMDIF=CSUMDIF/XIJMAX                                            TER0404
      NMAXM1=NMAX-1                                                     TER0406
      II=I-1                                                            TER0408
  262 PHI(1,1)=PHI(1,1)+ZMEANZ                                          TER0410
      DO 4001 I=1,IMAX                                                  TER0411
      DO 4001 J=1,JMAX                                                  TER0412
 4001 ZDIFF(I,J)=Z(I,J)-ZCOMP(I,J)                                      TER0413
      PRINT 4002                                                        TER0414
 4002 FORMAT(1H1////48H TABLE OF DIFFERENCES ACTUAL-COMPUTED ELEVATIONS TER0415
     1 ///)                                                             TER0416
      PRINT 2105,(J,J=1,JMAX)                                           TER0417
      DO 4003 I=1,IMAX                                                  TER0418
 4003 PRINT 2107,I,(ZDIFF(I,J),J=1,JMAX)                                TER0419
      IF(NEWCTL)  500,501,500                                           TER0420
C     EVALUATE GRID ELEVATIONS USING THE APPROXIMATING POLYNOMIAL TO BE TER0421
C     USED IN OVERLAY  3                                                TER0422
  500 X(1)=XFIRST+100. $ Y(1)=YFIRST+100.                               TER0423
      KK=AQ/25+1                                                        TER0424
      DO 502 I=2,KK $ X(I)=X(I-1)+25.                                   TER0425
  502 Y(I)=Y(I-1)+25.                                                   TER0426
      DO 550 I=1,KK $ DO 550 J=1,KK                                     TER0427
      ZCOMP(I,J)=0.                                                     TER0428
```

```
      IS=NMAX                                                            TER0429
  548 POLY=PHI(IS,MMAX)                                                  TER0430
      IT=MMAX-1                                                          TER0431
  549 POLY=POLY*(Y(J)-YMEANY)+PHI(IS,IT)                                 TER0432
      IF(IT-1) 2000,547,542                                              TER0433
  542 IT=IT-1                                                            TER0434
      GO TO 549                                                          TER0435
  547 ZCOMP(I,J)=ZCOMP(I,J)*(X(I)-XMEANX)+POLY                           TER0436
  543 IF(IS-1)2000,550,544                                               TER0437
  544 IS=IS-1                                                            TER0438
      GO TO 548                                                          TER0439
  550 CONTINUE                                                           TER0440
C  PRINT NEW DEPENDENT VARIABLES                                         TER0441
 3110 PRINT 551                                                          TER0442
  551 FORMAT(1H1////48HCOMPUTED GRID ELEVATIONS TO BE USED IN OVERLAY 3
     1 ///)                                                              TER0443
      PRINT 2105,(J,J=1,KK)                                              TER0444
      DO 560  I=1,KK                                                     TER0445
  560 PRINT 2107,I,(ZCOMP(I,J),J=1,KK)                                   TER0446
  501 PRINT 776                                                          TER0447
  776 FORMAT(1H1////40HCOEFFICIENTS OF APPROXIMATING POLYNOMIAL  ///)    TER0448
      PRINT 1272                                                         TER0449
 1272 FORMAT(     7H DEGREE 30X  14H DEGREE  OF  Y//      7H OF  X   )   TER0405
      PRINT 1273,(N,N=0,NMAXM1)                                          TER0450
 1273 FORMAT( 10X,3I20)                                                  TER0407
      DO 302 J=1,MMAX                                                    TER0451
      JJ=J-1                                                             TER0452
      PRINT 270,JJ ,(PHI(I,J),I=1,NMAX)                                  TER0453
  270 FORMAT(//I5,5X,3E20.10/(10X,3E20.10))                              TER0409
  302 CONTINUE                                                           TER0454
      PRINT 1190                                                         TER0455
 1190 FORMAT (////91H  MEAN SQUARED        COMPUTED MEAN SQUARED         TER0456
     1MAXIMUM DIFFERENCES                                    ///)        TER0457
      PRINT 1270,XMINSQD,CMINSQD,CMAXDIF                                 TER0458
 1270 FORMAT (E15.10,5X,E20.10, 7X,E20.10///)                            TER0459
      NMAXX=NMAX-1 $ MMAXX=MMAX-1
```

```
      PUNCH 778, NMAXX, MMAXX
  778 FORMAT(2I10)
      PUNCH 777,((PHI(I,J),I = 1,NMAX),J = 1,MMAX)                    TER0460
  777 FORMAT(4E20.10)                                                 TER0461
      GO TO 7777                                                      TER0462
 2000 PRINT 2001                                                      TER0463
 2001 FORMAT (  6H ERROR)                                             TER0464
      STOP                                                            TER0465
 1999 PRINT 1990                                                      TER0466
 1990 FORMAT(15H IT ERROR STOP     )                                  TER0467
      STOP                                                            TER0468
 7777 CONTINUE                                                        TER0469
      END                                                             TER0470
        FINIS                                                         TER0471
-BINARY,56.                                                           TER0472
(OVERLAY,7,3.                                                         TER0473
-FTN,L,E.                                                             TER0474
      PROGRAM PLOTTER                                                 TER0475
C     PLOTS A MAP USING THE APPROXIMATING POLYNOMIAL.
      COMMON/PLOT1/CNL,CND,NC,LABL(50),ITITLE(12)                     TER0476
     1      /PLOT2/ZGA(45,45),A,KK                                    TER0477
      DIMENSION ZG(45,45)                                             TER0478
      DO 10 I=1,KK $ DO 10 J=1,KK $ K=KK+1-J                          TER0479
   10 ZG(I,K)=ZGA(I,J)                                                TER0480
      CALL CONTOUR(KK,KK,ZG,45,45,CNL,CND,NC,LABL,ITITLE,08.,08.,0) . TER0483
      END                                                             TER0484
      SUBROUTINE CONTOUR(MA,NA,AM,MX,NY,CNL,CND,NK,IB,ITITLE,XSC,YSC, TER0485
     1KRID)                                                           TER0486
      DIMENSION AM(MX,NY) ,IT(12),X(1800),Y(1800),IB(50)              TER0487
      DIMENSION  YMIN(180),  XYMAX(180),ITITLE(12)                    TER0488
      COMMON IBODE,LOCMAX,IBLANK,IARGG,IGRID,FLND,FLMD,YSCALE,EXSCALE, TER0489
     1  IT,IBEL,MODCURV,ND,MD,XYMAX,YMIN,CL,Y,X                       TER0490
      MD=XMINOF(60,MA)                                                TER0491
      MD=XMAXOF(2,MD)                                                 TER0492
      ND=XMINOF(60, NA)                                               TER0493
      ND=XMAXOF(2,ND)                                                 TER0494
```

```
      FLMD=MD                                                           TER0495
      FLND=ND                                                           TER0496
      IF (IB) 51,50,51                                                  TER0497
   50 IQ = -1                                                           TER0498
      GO TO 52                                                          TER0499
   51 IQ = 0                                                            TER0500
   52 IF(NK-50) 18,18,19                                                TER0501
   19 NC = 50                                                           TER0502
      GO TO 24                                                          TER0503
   18 NC=NK                                                             TER0504
   24 IF(XSC-1.)2,1,1                                                   TER0505
    2 EXSCALE=INTF(FLMD/9.+1.)                                          TER0506
      IF(FLND/EXSCALE-15.) 3,3,4                                        TER0507
    3 YSCALE=EXSCALE                                                    TER0508
      GO TO 5                                                           TER0509
    4 YSCALE=INTF(FLND/15.+1.)                                          TER0510
      EXSCALE=YSCALE                                                    TER0511
      GO TO 5                                                           TER0512
    1 EXSCALE=MIN1F(30.,INTF(XSC))                                      TER0513
      IF(YSC   -1.) 2,16,16                                             TER0514
   16 YSCALE = MIN1F(30.,INTF(YSC))                                     TER0515
      IF(FLMD/EXSCALE-9.) 10,10,2                                       TER0516
   10 IF(FLND/YSCALE-15.) 5,5,2                                         TER0517
    5 IF(CND) 6,7,6                                                     TER0518
    6 CL=CNL                                                            TER0519
      CNDY = CND                                                        TER0520
      GO TO 12                                                          TER0521
    7 AMAX=-1.E 250                                                     TER0522
      AMIN=1.E250                                                       TER0523
      DO 8 I=1,MD                                                       TER0524
      DO 8 J=1,ND                                                       TER0525
      AA=AM(I,J)                                                        TER0526
      IF(AMAX-AA) 15,9,9                                                TER0527
   15 AMAX=AA                                                           TER0528
      GO TO 8                                                           TER0529
    9 IF(AMIN-AA) 8,8,11                                                TER0530
```

```
   11 AMIN=AA                                                    TER0531
    8 CONTINUE                                                   TER0532
      CNDY =  (AMAX-AMIN) / FLOATF (NC)                          TER0533
      CL=AMIN+.5*CNDY                                            TER0534
   12 IGRID=KRID                                                 TER0535
      IBLANK=8H                                                  TER0536
      IF(EXSCALE-5.) 20,20,21                                    TER0537
   20 IF(YSCALE-5.) 22,22,21                                     TER0538
   21 IQ=-1                                                      TER0539
   22  IBEL = IBLANK                                             TER0540
      KBK=0                                                      TER0541
      DO 40 I = 1,12                                             TER0542
      IF (ITITLE(I)) 42,41,42                                    TER0543
   41 IT(I) = IBLANK                                             TER0544
      KBK=KBK+1                                                  TER0545
      GO TO 40                                                   TER0546
   42 IT(I) = ITITLE(I)                                          TER0547
   40 CONTINUE                                                   TER0548
      IF(KBK-12) 60,61,61                                        TER0549
   61 IT(1)=8HNO NAME                                            TER0550
   60 XLEN = EXSCALE/24.                                         TER0551
      YLEN = YSCALE/24.                                          TER0552
      MODCURV=1                                                  TER0553
       IARGG=0                                                   TER0554
      DO 13 KLM = 1,NC                                           TER0555
      IF(IQ) 45,46,45                                            TER0556
   46 IBEL=IB(KLM)                                               TER0557
   45 CALL SCAN (AM,MD,ND,MX,NY,CL)                              TER0558
      CL = CL+CNDY                                               TER0559
   13 IARGG=1                                                    TER0560
C                                                                TER0561
      FLT1=FLMD+1.                                               TER0562
      FLT2 =   FLND + 1.                                         TER0563
      X(1)=0.                                                    TER0564
      Y(1)=FLT2                                                  TER0565
      X(2)=FLT1                                                  TER0566
```

```
      Y(2)=FLT2                                                        TER0567
      X(3)=FLT1                                                        TER0568
      Y(3)=0.                                                          TER0569
      IBODE=-1                                                         TER0570
      CALL PLOT (3)                                                    TER0571
      DO 14 KLM=1,LOCMAX                                               TER0572
      XXX=XYMAX(KLM) +XLEN/4.                                          TER0573
      YYY=YMIN(KLM)  +YLEN/4.                                          TER0574
      X(1) = XXX                                                       TER0575
      Y(1)  = YYY + YLEN                                               TER0576
      X(2) = XXX                                                       TER0577
      Y(2)  = YYY - YLEN                                               TER0578
      X(3) = XXX                                                       TER0579
      Y(3)  = YYY                                                      TER0580
      X(4) = XXX + XLEN                                                TER0581
      Y(4)  = YYY                                                      TER0582
      X(5) = XXX - XLEN                                                TER0583
      Y(5)  = YYY                                                      TER0584
   14 CALL PLOT (5)                                                    TER0585
      MODCURV=3                                                        TER0586
       X(1)=1.                                                         TER0587
       Y(1)=FLND                                                       TER0588
      X(2)=FLMD                                                        TER0589
      Y(2)=FLND                                                        TER0590
      X(3)=FLMD                                                        TER0591
      Y(3)=1.                                                          TER0592
      IF(IGRID) 32,31,32                                               TER0593
   32 NX = 3                                                           TER0594
      GO TO 33                                                         TER0595
   31  X(4)=1.                                                         TER0596
       Y(4)=1.                                                         TER0597
      X(5) = 1.                                                        TER0598
      Y(5) = FLND                                                      TER0599
      NX = 5                                                           TER0600
   33 CALL PLOT (NX)                                                   TER0601
      RETURN                                                           TER0602
```

```
      END                                                               TER0603
C                                                                       TER0604
      SUBROUTINE SCAN (AM, M, N, MX, NY, CL)                            TER0605
C THIS PROGRAM IS WRITTEN BY M.O.DAYHOFF                                TER0606
      DIMENSION AM(MX,NY),REC(900),  X(1800), Y(1800)                   TER0607
      DIMENSION IPT(3,3),INX(8),INY(8)                                  TER0608
      DIMENSION DUMDUM(384)                                             TER0609
      COMMON IBODE,DUMDUM,CV,Y,X,REC,THE,RA,DL,INY,INX,IPT,PY,JT,       TER0610
     1   NQ,NP,IV,IT,ISS,IDY,IDX,IY,IX,NI,NT,MT                         TER0611
      D=0.                                                              TER0612
      R=1.                                                              TER0613
      TH=1.570796325                                                    TER0614
      NP=0                                                              TER0615
      DL=D                                                              TER0616
      RA=R                                                              TER0617
      THE=TH                                                            TER0618
      MT=M                                                              TER0619
      NT=N                                                              TER0620
      CV=CL                                                             TER0621
      IF(IZW-120631) 1,3,1                                              TER0622
    1 IPT(1,1)=8                                                        TER0623
      IPT(1,2)=1                                                        TER0624
      IPT(1,3)=2                                                        TER0625
      IPT(2,1)=7                                                        TER0626
      IPT(2,3)=3                                                        TER0627
      IPT(3,1)=6                                                        TER0628
      IPT(3,2)=5                                                        TER0629
      IPT(3,3)=4                                                        TER0630
      INX(1)=-1                                                         TER0631
      INX(2)=-1                                                         TER0632
      INX(3)=0                                                          TER0633
      INX(4)=1                                                          TER0634
      INX(5)=1                                                          TER0635
      INX(6)=1                                                          TER0636
      INX(7)=0                                                          TER0637
      INX(8)=-1                                                         TER0638
```

```
      INY(1)=0                                                          TER0639
      INY(2)=1                                                          TER0640
      INY(3)=+1                                                         TER0641
      INY(4)=+1                                                         TER0642
      INY(5)=0                                                          TER0643
      INY(6)=-1                                                         TER0644
      INY(7)=-1                                                         TER0645
      INY(8)=-1                                                         TER0646
      IZW=120631                                                        TER0647
    3 XT=MT                                                             TER0648
      DO 58 J=1,900                                                     TER0649
   58 REC(J)=0                                                          TER0650
      ISS=0                                                             TER0651
    2 MT1=MT-1                                                          TER0652
      DO 110 I=1,MT1                                                    TER0653
      IF (AM(I,1)-CV) 55,110,110                                        TER0654
   55 IF (AM(I+1,1)-CV)110,57,57                                        TER0655
   57 IX=I+1                                                            TER0656
      IY=1                                                              TER0657
      IDX=-1                                                            TER0658
      IDY=0                                                             TER0659
      CALL TRACE (AM,MX, NY)                                            TER0660
  110 CONTINUE                                                          TER0661
      NT1=NT-1                                                          TER0662
      DO 20 I=1,NT1                                                     TER0663
      IF (AM(MT,I)-CV) 15,20,20                                         TER0664
   15 IF (AM(MT,I+1)-CV) 20,17,17                                       TER0665
   17 IX=MT                                                             TER0666
      IY=I+1                                                            TER0667
      IDX=0                                                             TER0668
      IDY=-1                                                            TER0669
      CALL TRACE (AM,MX, NY)                                            TER0670
   20 CONTINUE                                                          TER0671
   22 DO 30 I=1,MT1                                                     TER0672
      MT2=MT+1-I                                                        TER0673
      IF (AM(MT2,NT)-CV) 25,30,30                                       TER0674
```

```
25 IF (AM(MT2-1,NT)-CV) 30,27,27                                TER0675
27 IX=MT2-1                                                     TER0676
   IY=NT                                                        TER0677
   IDX=1                                                        TER0678
   IDY=0                                                        TER0679
   CALL TRACE (AM,MX, NY)                                       TER0680
30 CONTINUE                                                     TER0681
   DO 40 I=1,NT1                                                TER0682
   NT2=NT+1-I                                                   TER0683
   IF (AM(1,NT2)-CV) 35,40,40                                   TER0684
35 IF (AM(1,NT2-1)-CV) 40,37,37                                 TER0685
37 IX=1                                                         TER0686
   IY=NT2-1                                                     TER0687
   IDX=0                                                        TER0688
   IDY=1                                                        TER0689
   CALL TRACE (AM,MX, NY)                                       TER0690
40 CONTINUE                                                     TER0691
   ISS=1                                                        TER0692
   NT1=NT-1                                                     TER0693
   MT1=MT-1                                                     TER0694
   DO 10 J=2,NT1                                                TER0695
   DO 10 I=1,MT1                                                TER0696
   IF (AM(I,J)-CV) 5,10,10                                      TER0697
 5 IF (AM(I+1,J)-CV) 10,7,7                                     TER0698
 7 COM=100*(I+1)+J                                              TER0699
   IF (NP) 12,11,12                                             TER0700
12 DO 9 ID= 1,NP                                                TER0701
   IF (REC(ID)-COM) 9,10,9                                      TER0702
 9 CONTINUE                                                     TER0703
11 IX= I+1                                                      TER0704
   IY=J                                                         TER0705
   IDX=-1                                                       TER0706
   IDY=0                                                        TER0707
   CALL TRACE (AM,MX, NY)                                       TER0708
10 CONTINUE                                                     TER0709
   RETURN                                                       TER0710
```

```
      END                                                               TER0711
      SUBROUTINE TRACE (AM,MY,NY)                                       TER0712
      DIMENSION AM(MY,NY),REC(900),  X(1800), Y(1800)                   TER0713
      DIMENSION IPT(3,3), INX(8),INY(8)                                 TER0714
      DIMENSION DUMDUM(384)                                             TER0715
      COMMON IBODE,DUMDUM,CV,Y,X,REC,THE,RA,DL,INY,INX,IPT,PY,JT,       TER0716
     1 N   ,NP,IV,IT,ISS,IDY,IDX,IY,IX,NI,NT,MT                         TER0717
      PY=0.0                                                            TER0718
      RC=  COSF(THE)*RA                                                 TER0719
      RS=  SINF(THE)*RA                                                 TER0720
  501 JT=0                                                              TER0721
      N=0                                                               TER0722
      IX0=IX                                                            TER0723
      IY0=IY                                                            TER0724
      ISX=IDX+2                                                         TER0725
      ISY=IDY+2                                                         TER0726
      IS=IPT(ISX,ISY)                                                   TER0727
      JTB=0                                                             TER0728
      IS0=IS                                                            TER0729
      IF(IS0-8)18,18,17                                                 TER0730
   17 IS0=IS0-8                                                         TER0731
   18 IT=0                                                              TER0732
    5 CONTINUE                                                          TER0733
      CALL CALC  (AM,MY,NY)                                             TER0734
      NZ=N                                                              TER0735
      N=NZ                                                              TER0736
      IF (IT+JT-1) 49,49,47                                             TER0737
   47 XS=X(N-1)                                                         TER0738
      YS=Y(N-1)                                                         TER0739
      X(N-1)=X(N)                                                       TER0740
      Y(N-1)=Y(N)                                                       TER0741
      X(N)=XS                                                           TER0742
      Y(N)=YS                                                           TER0743
   49 IS=IS+1                                                           TER0744
      JT=IT                                                             TER0745
    9 IF (IS-9) 8,7,7                                                   TER0746
```

```
      7 IS=IS-8                                                           TER0747
      8 IDX=INX(IS)                                                       TER0748
        IDY=INY(IS)                                                       TER0749
        IX2=IX+IDX                                                        TER0750
        IY2=IY+IDY                                                        TER0751
        JTB=JTB+1                                                         TER0752
        IF (JTB-1799) 51,51,308                                           TER0753
    308 PRINT                103,CV,X(N),Y(N)                             TER0754
    103 FORMAT(1H0,23HA CONTOUR LINE AT LEVEL,E12.5,21H WAS TERMINATED AT TER0755
       1X=,E12.5,3H Y=,E12.5/                                             TER0756
       2 48H BECAUSE IT CONTAINED MORE THAN 1799 PLOT POINTS  )           TER0757
        RETURN                                                            TER0758
C                                                                         TER0759
     51 IF (JTB-900) 5100,5101,5101                                       TER0760
   5100 IBODE = 0                                                         TER0761
        GO TO 5102                                                        TER0762
   5101 IBODE = 1                                                         TER0763
   5102 IF(ISS) 10,10,20                                                  TER0764
     20 IF(IX-IX0) 12,21,12                                               TER0765
     21 IF(IY-IY0) 12,22,12                                               TER0766
     22 IF(IS-IS0) 12,23,12                                               TER0767
     23 CONTINUE                                                          TER0768
        CALL CALC  (AM,MY,NY)                                             TER0769
        GO TO 73                                                          TER0770
     10 IF(IX2) 13,50,13                                                  TER0771
     13 IF (IX2-MT) 19,19,50                                              TER0772
     19 IF (IY2) 11,50,11                                                 TER0773
     11 IF (IY2-NT) 12,12,50                                              TER0774
     12 IF (CV-AM(IX2,IY2)) 206,206,5                                     TER0775
    206 IF (IDX**2+IDY**2-1) 213,6,213                                    TER0776
    213 DCP=(AM(IX,IY)+AM(IX2,IY)+AM(IX,IY2)+AM(IX2,IY2))/4.0             TER0777
        IF (DCP-CV) 5,217,217                                             TER0778
    217 IF (INX(IS-1)) 214,215,214                                        TER0779
    214 IX=IX+IDX                                                         TER0780
        IDX=-IDX                                                          TER0781
        PY=2.0                                                            TER0782
```

```
      CALL CALC   (AM,MY,NY)                                            TER0783
      IX=IX+IDX                                                         TER0784
      GO TO 6                                                           TER0785
  215 IY=IY+IDY                                                         TER0786
      IDY=-IDY                                                          TER0787
      PY=2.0                                                            TER0788
      CALL CALC   (AM,MY,NY)                                            TER0789
      IY=IY+IDY                                                         TER0790
    6 IF (AM(IX-1,IY) - CV) 306,16,16                                   TER0791
  306 NP=NP+1                                                           TER0792
      REC(NP)=100*IX+IY                                                 TER0793
   16 IS=IS+5                                                           TER0794
      IX=IX2                                                            TER0795
      IY=IY2                                                            TER0796
      GO TO 9                                                           TER0797
   50 XT=MT                                                             TER0798
      IF (AM(IX-1,IY)-CV) 307,73,73                                     TER0799
  307 NP=NP+1                                                           TER0800
      REC(NP)=100*IX+IY                                                 TER0801
   73 DO 74 I=1,N                                                       TER0802
      X(I)=X(I)+RC*Y(I)                                                 TER0803
   74 Y(I)=RS*Y(I)                                                      TER0804
      CALL PLOT (     N)                                                TER0805
      RETURN                                                            TER0806
      END                                                               TER0807
      SUBROUTINE CALC   (AM,MY,NY)                                      TER0808
      DIMENSION AM(MY,NY),REC(900),   X(1800), Y(1800)                  TER0809
      DIMENSION IPT(3,3),INX(8),INY(8)                                  TER0810
      DIMENSION DUMDUM(384)                                             TER0811
      COMMON IBODE,DUMDUM,CV,Y,X,REC,THE,RA,DL,INY,INX,IPT,PY,JT,       TER0812
     1 N    ,NP,IV,IT,ISS,IDY,IDX,IY,IX,NI,NT,MT                        TER0813
      IT=0                                                              TER0814
      N=N+1                                                             TER0815
      IF (IDX**2 + IDY**2 -1) 20,1,20                                   TER0816
    1 IF (IDX) 10,2,10                                                  TER0817
    2 X(N)=IX                                                           TER0818
```

```
      Z=IY                                                              TER0819
      IY2=IY+IDY                                                        TER0820
      DY=IDY                                                            TER0821
   41 Y(N) =  ((AM(IX,IY)-CV)/(AM(IX,IY)-AM(IX,IY2))) *DY + Z           TER0822
      RETURN                                                            TER0823
   10 Y(N)=IY                                                           TER0824
      W=IX                                                              TER0825
      DX=IDX                                                            TER0826
      IX2=IX+IDX                                                        TER0827
   44 X(N) =  ((AM(IX,IY)-CV)/(AM(IX,IY)-AM(IX2,IY))) *DX +W            TER0828
      RETURN                                                            TER0829
   20 IX2=IX+IDX                                                        TER0830
      IY2=IY+IDY                                                        TER0831
      W=IX                                                              TER0832
      Z=IY                                                              TER0833
      DX=IDX                                                            TER0834
      DY=IDY                                                            TER0835
      DCP=(AM(IX,IY)+AM(IX2,IY)+AM(IX,IY2)+AM(IX2,IY2))/4.0             TER0836
      IF (PY-2.0) 24,21,24                                              TER0837
   24 IF (DCP-CV) 21,21,25                                              TER0838
   21 AL=AM(IX,IY)-DCP                                                  TER0839
   23 V=.5*(AL+DCP -CV)/AL                                              TER0840
   27 X(N)=V*DX+W                                                       TER0841
      Y(N)=V*DY+Z                                                       TER0842
      PY=0.0                                                            TER0843
      RETURN                                                            TER0844
   25 IT=1                                                              TER0845
      AL=AM(IX2,IY2)-DCP                                                TER0846
   33 V=.5*(AL+DCP-CV)/AL                                               TER0847
   28 X(N)=-V*DX+W + DX                                                 TER0848
      Y(N)=-V*DY+Z + DY                                                 TER0849
      RETURN                                                            TER0850
      END                                                               TER0851
      SUBROUTINE PLOT  (NGG)                                            TER0852
      DIMENSION  X(1800),Y(1800),ITITLE(12),                            TER0853
     1          LTITLE(14), KAXIS(5), ICURV(460), JGRID(25), ICONT(1)TER0854
```

```
      DIMENSION XMAX(180),YMAX(180),XMIN(180),YMIN(180),XYMAX(180)
      COMMON IBODE,LOCMAX,IBLANK,IARGG,IGRID,FLND,FLMD,YSCALE,EXSCALE,
     1  ITITLE,LABEL,MODCURV,ND,MD,XYMAX,YMIN,CL,Y,X,DUMIE(942),ICURV
      DATA (ICONTRL=40000B),(ICURV3=3777377720202020B),(ICURV5=
     1  10400000000000B),(IMAS1=10100000000000B),(IMAS2=
     2  10400000000000B)
C
      NN=NGG
      ICURV4 = IMAS1
      IF (IBODE) 95,4449,67
 4449 IF (IARGG) 98,97,4
   97 LOCMAX=0
    4 FINX=X(NN)
      FINY=Y(NN)
      IF(ABSF(FINX-1.)-.00001)61,61,62
C LEFT MARGIN
   61 NN=NN+1
      X(NN) = 1. - .31*EXSCALE
      Y(NN)=FINY
      GOTO 63
   62 IF(ABSF(FINY-1.)-.00001) 69,69,65
C TOP MARGIN
   69 NN=NN+1
      X(NN)=FINX
      Y(NN)=FINY- YSCALE/45.
      GO TO 33
   65 IF(ABSF(FINX-FLMD)-.00001) 70,70,66
C RIGHT MARGIN
   70 NN=NN+1
      X(NN)=FINX+ EXSCALE/45.
      Y(NN)=FINY
      GO TO 63
   66 IF(ABSF(FINY-FLND)-.00001) 68,68,67
C BOTTOM MARGIN
   68 NN=NN+1
      X(NN)=FINX
```

```
      Y(NN) = FINY + .31 * YSCALE                                       TER0891
C WRITE LABEL VERTICALLY                                                TER0892
      GO TO 33                                                          TER0893
C WRITE LABEL HORIZONTALLY                                              TER0894
   63 ICURV4 = IMAS2                                                    TER0895
      GO TO 33                                                          TER0896
C INTERNAL CONTOUR                                                      TER0897
   67 IF(IARGG)  98,95,96                                               TER0898
   98 STOP                                                              TER0899
   96 CYMAX = AMX(NN,Y)                                                 TER0900
      CYMIN = AMN(NN,Y,IK)                                              TER0901
      CXYMAX=X(IK)                                                      TER0902
      CXMAX = AMX(NN,X)                                                 TER0903
      CXMIN = AMN (NN,X,IK)                                             TER0904
      DO 91 II = 1,LOCMAX                                               TER0905
      IK = II                                                           TER0906
      IF(CYMIN-YMAX(II)) 90,91,91                                       TER0907
   90 IF(CYMIN-YMIN(II)) 91,91,92                                       TER0908
   92 IF(CXYMAX-XMAX(II)) 93,91,91                                      TER0909
   93 IF(CXYMAX-XMIN(II)) 91,91,94                                      TER0910
   91 CONTINUE                                                          TER0911
      LOCMAX=LOCMAX+1                                                   TER0912
        IK=LOCMAX                                                       TER0913
   94 XMAX(IK)=CXMAX                                                    TER0914
      YMAX(IK)=CYMAX                                                    TER0915
      XMIN(IK)=CXMIN                                                    TER0916
      YMIN(IK)=CYMIN                                                    TER0917
      XYMAX(IK)=CXYMAX                                                  TER0918
   95 LABEL=IBLANK                                                      TER0919
      GO TO 330                                                         TER0920
   33 IF(LABEL-IBLANK) 330,331,330                                      TER0921
  331 NN = NN-1                                                         TER0922
  330 IF (IBODE) 81,81,82                                               TER0923
   82 NNN = NN-900                                                      TER0924
      NN = 900                                                          TER0925
C         PROGRAMMER   J. R. WARD                                       TER0926
```

```
C
   81 NUMPTS = NN
      IF(MODCURV-2) 10,2400,2400
 2400 BACKSPACE 49
      GO TO 240
   10 ICONT(1) = ICONTRL + 2
C            INSERT TITLE SIZE (02B) AHEAD OF MAIN TITLE RECORD.
      CALL ISHIFT6 (ITITLE, LTITLE)
C           TEST FOR ALL BLANK TITLES.
      DO 9075 I=1,6
      IF(ITITLE(I)-IBLANK) 9074,9075,9074
 9074 IF(ITITLE(I) ) 9080,9075,9080
 9075 CONTINUE
      IT1 = 1
      ICONT(1) = ICONT(1) - 1
      GO TO 9081
 9080 IT1 = 0
 9081 DO 9085 I=7,12
      IF (ITITLE(I) - IBLANK) 9084,9085,9084
 9084 IF (ITITLE(I))9090,9085,9090
 9085 CONTINUE
      IT2 = 1
      ICONT(1) = ICONT(1) - 1
      GO TO 9091
 9090 IT2 = 0
C            NOW GENERATE AXES RECORDS.
 9091 LFTMGN = 0
      IBOTMGN = 40
      XFR1 = FLMD / EXSCALE
      JXFR = XFR1
      XFR2 = (FLMD + 1.)/EXSCALE
      JXFR3 = (FLMD-1.)/EXSCALE
      YFR1 = FLND/YSCALE
      JYFR = YFR1
      YFR2 = (FLND+1.)/YSCALE
      IX100 = 100./EXSCALE
```

```
      IY100 = 100./YSCALE                                               TER0963
      IXAXIS=0                                                          TER0964
        IYAXIS = JYFR+1                                                 TER0965
      ISIZEX = 100. * XFR2                                              TER0966
      ISIZEY = 100. * YFR2                                              TER0967
      IH = LFTMGN                                                       TER0968
      JH = IBOTMGN +IYAXIS*100                                          TER0969
      LH = ISIZEX                                                       TER0970
      IHL = 100 * JXFR3 + IX100 - 7                                     TER0971
      KAXIS(1) = IPACK12(IH,JH,LH,IHL)                                  TER0972
      JHL = JH + 2                                                      TER0973
      IHL2 = -100                                                       TER0974
      IVH2 = -EXSCALE                                                   TER0975
      IVH = -IVH2 * JXFR3 + 1                                           TER0976
      KAXIS(2) = IPACK12(JHL,IHL2,IVH,IVH2)                             TER0977
      NH = JXFR3+ 1                                                     TER0978
      ISH = 8H      14                                                  TER0979
      IV = LFTMGN                                                       TER0980
      JV = IBOTMGN + IYAXIS*100 - ISIZEY                                TER0981
      KAXIS(3) = IPACK12(NH,ISH,IV,JV)                                  TER0982
      LV = ISIZEY                                                       TER0983
      IVL = IV - 2                                                      TER0984
      JVL = JV + LV -   7 -IY100                                        TER0985
      JVL2 = -100                                                       TER0986
      KAXIS(4) = IPACK12(LV,IVL,JVL,JVL2)                               TER0987
      IVV = 1                                                           TER0988
      IVV2 =  YSCALE                                                    TER0989
      INV = (FLND-1.)/YSCALE  + 1.                                      TER0990
      ISV = 8H      11                                                  TER0991
      KAXIS(5) = IPACK12(IVV,IVV2,INV,ISV)                              TER0992
C         NOW GENERATE CURVES.                                          TER0993
      SCX = 100./EXSCALE                                                TER0994
      SCY = 100./YSCALE                                                 TER0995
      SHIFTX = IH                                                       TER0996
      SHIFTY = JH                                                       TER0997
  240 IF(NUMPTS - 900)241,242,241                                       TER0998
```

```
  241 SAVE1 = X(NUMPTS + 1)                                   TER0999
      SAVE2 = Y(NUMPTS + 1)                                   TER1000
      X(NUMPTS + 1) = X(NUMPTS)                               TER1001
      Y(NUMPTS + 1) = Y(NUMPTS)                               TER1002
  242 INUM = (NUMPTS + 1)/2                                   TER1003
      DO 244 I=1,INUM                                         TER1004
      IC1= X(2*I-1)*SCX + SHIFTX                              TER1005
      IC2=-Y(2*I-1)*SCY + SHIFTY                              TER1006
      IC3= X(2*I)*SCX + SHIFTX                                TER1007
      IC4=-Y(2*I)*SCY + SHIFTY                                TER1008
      ICURV(I+1) = IPACK12(IC1,IC2,IC3,IC4)                   TER1009
  244 II = I + 3                                              TER1010
      IF(NUMPTS - 900)245,246,245                             TER1011
  245 X(NUMPTS + 1) = SAVE1                                   TER1012
      Y(NUMPTS + 1) = SAVE2                                   TER1013
  246 CALL IPACKL1(LABEL, LABEL1, IDUMMY)                     TER1014
      ICURV(II-1) = LABEL1                                    TER1015
      ICURV(II) = ICURV4                                      TER1016
C         NOW WRITE RECORDS.                                  TER1017
      IF(MODCURV - 1)1260, 1260, 9015                         TER1018
 1260 CALL IWRITE (ICONT, IDUMMY, ICONT)                      TER1019
      IF(IDUMMY)5000,260,5000                                 TER1020
  260 IF(IT1)9269,9268,9269                                   TER1021
 9268 CALL IWRITE (LTITLE, IDUMMY, LTITLE(7))                 TER1022
      IF(IDUMMY)5000,9269,5000                                TER1023
 9269 IF(IT2)9271,9270,9271                                   TER1024
 9270 CALL IWRITE (LTITLE(8), IDUMMY, LTITLE(14))             TER1025
      IF(IDUMMY)5000,9271,5000                                TER1026
 9271 CALL IWRITE (KAXIS, IDUMMY, KAXIS(5))                   TER1027
      IF(IDUMMY)5000,9015,5000                                TER1028
 9015 CALL IWRITE (ICURV, IDUMMY, ICURV(II))                  TER1029
      IF(IDUMMY)5000,9020,5000                                TER1030
 9020 IF(MODCURV - 1)272,272,9025                             TER1031
  272 IF(IGRID - 1)9025,273,9025                              TER1032
C         GENERATE GRID IF CALLED FOR.                        TER1033
  273 IBOT = JV + IY100                                       TER1034
```

```
      ITOP = JH - IY100                                                 TER1035
      IRHT = LH - IX100                                                 TER1036
      NEXT1 = ITOP                                                      TER1037
      NEXT2 = IRHT                                                      TER1038
      JGRID(1) = 0                                                      TER1039
      DO 1274 J=1,19,2                                                  TER1040
      JGRID(J+1) = IPACK12 (IX100 , NEXT1, NEXT2, NEXT1)                TER1041
      IQG = NEXT1- IBOT  -100                                           TER1042
      IF (IQG) 1275,1275,1290                                           TER1043
 1290 IF (XABSF(IQG)-1) 1275,1275,1273                                  TER1044
 1273 NEXT1 = NEXT1 - 100                                               TER1045
      JGRID(J+2) = IPACK12 (NEXT2, NEXT1, IX100 , NEXT1)                TER1046
      IQG = NEXT1- IBOT  -100                                           TER1047
      IF (IQG) 1276,1276,1291                                           TER1048
 1291 IF (XABSF(IQG)-1) 1276,1276,1274                                  TER1049
 1274 NEXT1 = NEXT1 - 100                                               TER1050
 1275 JGRID(J+2) = IPACK12 (NEXT2, NEXT1, NEXT2, NEXT1)                 TER1051
 1276 JGRID(J+3) = ICURV3                                               TER1052
      JGRID(J+4) = ICURV5                                               TER1053
      CALL IWRITE (JGRID, IDUMMY, JGRID(J+4))                           TER1054
      IF(IDUMMY)5000,1277,5000                                          TER1055
 1277 NEXT1 = IX100                                                     TER1056
      NEXT2 = ITOP                                                      TER1057
      DO 1279 J=1,11,2                                                  TER1058
      JGRID(J+1) = IPACK12 (NEXT1, IBOT ,   NEXT1, NEXT2)               TER1059
      IQG = NEXT1 - IRHT +100                                           TER1060
      IF (IQG) 1292,1280,1280                                           TER1061
 1292 IF (XABSF(IQG)-1) 1280,1280,1278                                  TER1062
 1278 NEXT1 = NEXT1 + 100                                               TER1063
      JGRID(J+2) = IPACK12 (NEXT1, NEXT2, NEXT1, IBOT    )              TER1064
      IQG = NEXT1 - IRHT +100                                           TER1065
      IF(IQG) 1293,1281,1281                                            TER1066
 1293 IF(XABSF(IQG)-1) 1281,1281,1279                                   TER1067
 1279 NEXT1 = NEXT1 + 100                                               TER1068
 1280 JGRID(J+2) = IPACK12 (NEXT1, NEXT2, NEXT1, NEXT2)                 TER1069
 1281 JGRID(J+3) = ICURV3                                               TER1070
```

```
      JGRID(J+4) = ICURV5                                               TER1071
      CALL IWRITE (JGRID, IDUMMY, JGRID(J+4))                           TER1072
      IF(IDUMMY)5000,9025,5000                                          TER1073
C         SET UP RETURN.                                                TER1074
 9025 IF (MODCURV-3) 87,278,87                                          TER1075
  278 PRINT 130, (ITITLE(I),I=1,12)                                     TER1076
  130 FORMAT(21H0CONTOUR GRAPH TITLED/2(5X,6A8)/18H HAS BEEN PLOTTED./  TER1077
     11H0)                                                              TER1078
      IDUMMY = ITYP2(IDUMMY)                                            TER1079
      GO TO 888                                                         TER1080
C         THESE ARE THE NORMAL RETURNS.                                 TER1081
C         NOW SET UP THE RETURN FOLLOWING A TAPE ERROR.                 TER1082
 5000 IF(MODCURV - 1)5001,5001,5002                                     TER1083
 5001 IDUMMY = ITYPE1(IDUMMY)                                           TER1084
      GO TO 1260                                                        TER1085
 5002 PRINT 5100                                                        TER1086
 5100 FORMAT (/, 36H TAPE ERROR IN WRITING GRAPH OUTPUT. )              TER1087
      IDUMMY = ITYPE1(IDUMMY)                                           TER1088
      STOP                                                              TER1089
C                                                                       TER1090
   87 MODCURV=2                                                         TER1091
  888 END FILE 49                                                       TER1092
      IF (EOF,49) 5000,88                                               TER1093
   88 IF (IBODE) 83,83,84                                               TER1094
   84 IBODE = 0                                                         TER1095
      NN = NNN                                                          TER1096
      DO 85 I = 1,NN                                                    TER1097
      II = I+900                                                        TER1098
      X(I) = X(II)                                                      TER1099
   85 Y(I) = Y(II)                                                      TER1100
      GO TO 4                                                           TER1101
C                                                                       TER1102
   83 RETURN                                                            TER1103
      END                                                               TER1104
C                                                                       TER1105
      FUNCTION AMX(KK,A)                                                TER1106
```

```
      DIMENSION A(1800)                                                 TER1107
      F=A(1)                                                            TER1108
      DO 1 II=2,KK                                                      TER1109
      AA=A(II)                                                          TER1110
      IF(F-AA) 2,1,1                                                    TER1111
    2 F=AA                                                              TER1112
    1 CONTINUE                                                          TER1113
    4 AMX=F                                                             TER1114
      RETURN                                                            TER1115
      END                                                               TER1116
      FUNCTION  AMN(KK,A,IK)                                            TER1117
      DIMENSION A(1800)                                                 TER1118
      F=A(1)                                                            TER1119
      IK = 1                                                            TER1120
      DO 1 II=2,KK                                                      TER1121
      AA=A(II)                                                          TER1122
      IF(AA-F) 2,1,1                                                    TER1123
    2 F=AA                                                              TER1124
      IK=II                                                             TER1125
    1 CONTINUE                                                          TER1126
      AMN=F                                                             TER1127
      RETURN                                                            TER1128
      END                                                               TER1129
C                                                                       TER1130
      SUBROUTINE IWRITE(IFIRST, IDUMMY, LAST)                           TER1131
C IFIRST IS THE FIRS VARIABLE TO BE WRITTEN AND ALL VARIABLES FROM      TER1132
C   IT TO LAST WILL BE WRITTEN IN BINARY MODE ON TAPE XX                TER1133
C         IF RECORD CORRECTLY WRITTEN, OTHERWISE SET NON-ZERO. NOTE THATTER1134
      I=3                                                               TER1135
 1000 BUFFER OUT(49,1)(IFIRST,LAST)                                     TER1136
C  TEST FOR WRITE ERRORS AND EOT                                        TER1137
    2 IF (UNIT,49)2,6,4,5                                               TER1138
C EOT OCCURRED                                                          TER1139
    4 IDUMMY=10                                                         TER1140
      GO TO 9                                                           TER1141
C  TAPE WRITE ERROR                                                     TER1142
```

```
    5 BACKSPACE 49
      IF(I) 4,4,66
   66 I=I-1
      GO TO 1000
C  NO ERROR NOEOT, WRITE OKAY
    6 IDUMMY=0
    9 RETURN
    8 IDUMMY=IDUMMY+10
      END
C
      FUNCTION ITYP2 (IDUMMY)
C    TYPE WORD -CONTOUR-
      DATA(IGRAPH=7HCONTOUR)
       WRITE(45,1)IGRAPH
      ITYP2=IDUMMY
    1 FORMAT(/ A7)
      RETURN
      END
C
      FUNCTION ITYPE1 (IDUMMY)
C          REWIND TAPE 8, REQUEST NEW TAPE, AND WAIT TILL READY.
      DIMENSION   IOUT( 4   )
      DATA ( IOUT=8H PLEASE ,8HLOAD NEW,8H TAPE 8.,8H START.  )
      END FILE 49
      REWIND 49
       WRITE(45,1)IOUT
    1 FORMAT(/ 4A8)
      ITYPE1=IDUMMY
      PAUSE  77777
      RETURN
      END
C
C
      SUBROUTINE ISHIFT6(ITITLE,LTITLE)
C          INSERTS 02B AHEAD OF 6-WORD TITLE RECORD.
      DIMENSION ITITLE(12),LTITLE(14)
```

```
      ICON2=1H2
      IBLANK=8H
      ENCODE(56,1,LTITLE) ICON2, (ITITLE(I),I=1,6),IBLANK
    1 FORMAT (A1,6A8,A7 )
      ENCODE(56,1,LTITLE(8)) ICON2,(ITITLE(I),I=7,12),IBLANK
      RETURN
      END
C
      FUNCTION IPACK12(IONE,I2,I3,I4)
C           PACKS FOUR 12-BIT WORDS INTO ONE 48-BIT WORD.
      DATA(ITWOTO35=1000000000000B),(ITWOTO23=100000000B),(ITWOTO11=1000
     10B),(IZER12=7777B)
       I11=IONE.AND.IZER12
      I22=I2.AND.IZER12
      I33=I3.AND.IZER12
      I22=I22*ITWOTO23
      I33=I33*ITWOTO11
       I11=I11*ITWOTO35
      IPACK12=I11.OR.I22.OR.I33.OR.I4.AND.IZER12
      RETURN
      END
C
      SUBROUTINE IPACKL1(LABEL1,LABEL2,LABEL3)
C  UNPACKS AND REPACKS  2 FOUR CHARACTER LABELS
      DATA(FLAG1=37773777000000000B)
      ENCODE(8,1,LABEL2)LABEL1
    1 FORMAT(4X,A4)
      LABEL2=FLAG1.OR.LABEL2
      ENCODE(8,2,LABEL3)LABEL1
    2 FORMAT(4X,R4)
      LABEL3=FLAG1.OR.LABEL3
      RETURN
      END
         END
          FINIS
-EXECUTE.
```

APPENDIX V

Inputs and Outputs of the Terrain Simulation Program

The following information is presented for potential users of this program:

A. Tape units

(1) The COOP control cards are shown in the FORTRAN listing (See Appendix IV.) A dash (-) in column one indicates a 7,9 punch, while an open parenthesis (() indicates an 11,0,7,9 punch.

(2) Logical tape unit 49 is the graph tape and 45 refers to the typewriter.

(3) Scratch tapes 56 and 57 are necessary and therefore a total of eight tape units are required.

B. Input Formats

The preparation of the input data cards is explained on the following pages of this appendix.

Data Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
1	2	ITITLE	1-48	These two cards contain the two line title desired for the first page of the output and the bottom of the plotted map.
2	1	NL	1-5	A right justified integer, the number of contour lines requiring a label.
3	NL	II	1-5	A right justified integer, the number of the contour level to be labeled, $1 \leq II \leq 50$
		LABL(II)	7-10	A left justified integer, the elevation of contour level II.
4	1	NMAX	1-5	A right justified integer, the maximum degree of x to be considered
		MMAX	6-10	A right justified integer, the maximum degree of y to be considered.
		NEWCTL	11-15	A right justified integer, NEWCTL = 0, no contour map is desired. = 1, an output tape will be generated for the CalComp 165 plotter.

Data Group	Number of Cards	Variable Name	Columns of Cards	Description of Variable
5	1	S	1-10	A right justified floating point number, the spacing desired between grid lines. S must be chosen so that neither the number of vertical nor horizontal grid lines exceed 45.
		A	11-20	A right justified floating point number, the linear dimension of the map square.
6	1	XTMIN	1-10	A right justified floating point number, the elevation of the lowest contour in the map square.
		XTMAX	11-20	A right justified floating point number, the maximum elevation in the map square.
		CND	21-30	A right justified floating point number, the contour interval. The first contour is the value C = XTMIN.
7	1	GX	1-2	The first two digits of a five digit floating point number, GX * L000 = the x coordinate of the lower left corner of the map square.

Data Group	Number of Cards	Variable Name	Columns of Cards	Description of Variation
		GY	3-4	The first two digits of a five digit floating point number, GY * 1000 = the y coordinate of the lower left corner of the map square.
		XP (1)	11-13	A right justified floating point number, the x coordinate of the lower left corner of the map square measured on an arbitrary scale. XP (1) usually should equal 0000.
		YP(1)	16-20	The y coordinate of the lower left corner measured on an arbitrary scale. YP(1) usually should equal 0000.
		XP(2)	21-25	The x coordinate of the lower right corner,
		YP(2)	26-30	The y coordinate of the lower right corner.
		XP (3)	31-35	The x coordinate of the upper right corner.
		XP(3)	36-40	The y coordinate of the upper right corner.
		XP(4)	41-45	The x coordinate of the upper left corner.
		XP(4)	46-50	The y coordinate of the upper left corner.

Data Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
8	As many cards as necessary		1-4	Any identifying code desired Col. 1-4 are not read.
		FZ	5-8	A right justified floating point number, FZ * 10 = the elevation (feet) whose coordinates appear on this card.
		NC	9-10	A sequence number starting at 01 and increasing by one on each card following whose coordinates are those of the same contour line.
		X(1)	11-15	A right justified floating point number, the x coordinate of a point on the contour line.
		Y(1)	16-20	A right justified floating point number, the coordinate of a point on the contour line.
			21-80	F 5.0 fields of additional points on the contour line in the same manner as Col. 11-20.

NOTE: One blank card is inserted after this group of data cards (eighth data group) for each contour line. One additional blank card is inserted if a new seventh data group is read.

Data Group	Number of Cards	Variable Name	Columns of Card	Description of Variable
9	1		7-9	The figure 999 should appear in these colums to signify that this is the last card of the data deck.

NOTE: There should be no blank cards between the last eighth data group and the ninth data group.

Example:

It is desired that grid square 7175 of the U. S. Army Hunter-Liggett Military Reservation be approximated by a polynomial. Figure 17 illustrates the data cards necessary.

The words

HLMR CLOSE TERRAIN

GRID SQUARE 7175

will appear at the top of the first page of the printed output and at the bottom of the plotted map. The minimum and maximum elevation appearing in this square are 900 and 1200 feet respectively. The contour interval is 20 feet. The maximum degree of X and Y to be considered is 20. The grid interval will be 50 meters and the square is 1000 meters on a side.

Figure 16 illustrates how the original map was enlarged to read the contour coordinates.

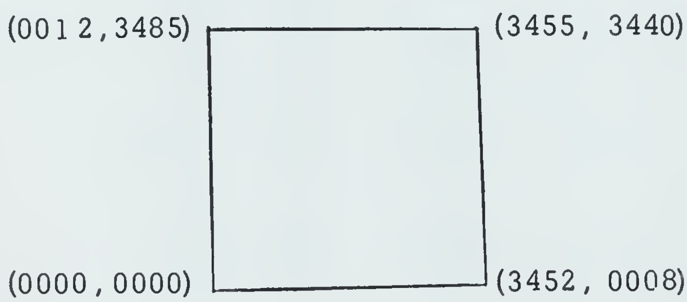


FIGURE 16

DISTORTION CAUSED BY PROJECTION
OF A MAP GRID SQUARE

Distortion caused by projection is apparent in Figure 16 and is corrected by the first overlay. The corner coordinates are arbitrary and all contour coordinates are relative to them. If the points are measured by hand, the actual corner coordinates may be used and in this case would

```
HLMR CLOSE TERRAIN
GRID SQUARE 7175
     3
     1  900
     6  1000
    11  1100
    20    20      1
          50         1000
         900         1200           20
7175          0000 0000 3452 0008 3455 3440 0012 3485
7175 09001 2799-0156 2864-0122 2889-0097 2894-0082 2889-0037 2862 0013 2849 0054
7175    02 2852 0075 2852 0118 2847 0133 2810 0173 2780 0215 2782 0233 2796 0247

7175 10201 3390 0021 3384-0013 3365-0019 3345-0024 3334-0012 3317 0022 3309 0056
7175    02 3307 0097 3313 0119 3317 0151 3326 0168 3337 0174 3350 0171 3367 0163


7175          0000 0000 3444 0020 3441 3467-0004 3475
7175 12001 2146 3552 2120 3499 2104 3476 2107 3449 2121 3433 2141 3431 2170 3431
7175    02 2192 3450 2228 3484 2263 3508 2302 3541 2322 3548 2341 3551 2360 3547
     999
```

FIGURE 17

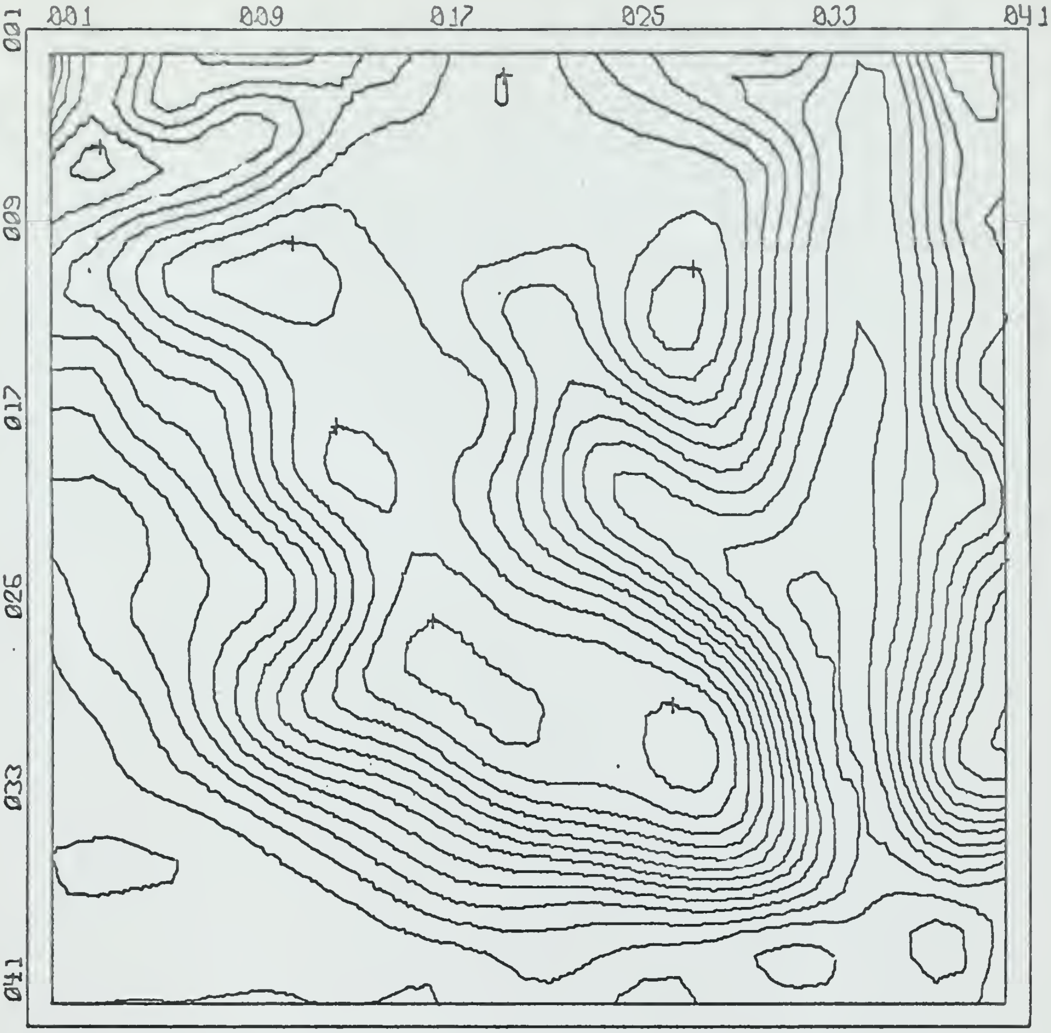
A SAMPLE DATA DECK FOR THE TERRAIN PROGRAM

be (71000, 75000), (72000, 75000), (72000, 76000), (71000, 76000). The contour coordinates would then be the actual terrain coordinates.

In this example the grid square was reprojected a second time with new corner coordinates of (0000, 0000), (3444,-0020), (3441, 3467), (-0004, 3475).

Terrain has three types of output, a) printed, b) punched cards, and c) a plotted map (See Figure 18). The printed output consists of 1) the x and y coordinates of the grid to be used, 2) the input elevations at points on this grid as interpolated by Overlay 1, 3) the elevations of the same points as computed using the approximating polynomial, 4) a table of differences between the actual and computed elevations of the grid, 5) the computed elevations to be used in Overlay 3, 6) the coefficients of the approximating polynomial, and 7) the actual mean elevation squared, computed mean elevation squared and the maximum differences between the actual and computed elevations. Although accuracy is not determined in general, if the actual mean elevation squared and computed mean elevation squared are equal, then computation is exact. In practice they will not be equal due to the imprecise nature of calculation. A wide discrepancy indicates excessive errors in calculation.

The first card of the punched card output contains the degree of x and y in a 2I10 format. The remaining cards contain the coefficients of the approximating polynomial in a 4E20.10 format.



HLMR CLOSE TERRAIN
GRID SQUARE 7174

FIGURE 18

A SAMPLE OF A PLOTTED MAP
FOR THE TERRAIN PROGRAM